



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

Incorporando a Edição de Competências a uma Rede Social Descentralizada para Utilização em Local de Trabalho

Norberto Luz Caires Junior

Monografia apresentada como requisito parcial
para conclusão do Curso de Computação — Licenciatura

Orientadora

Prof.a Dr.a Germana Menezes da Nóbrega

Brasília
2019



Incorporando a Edição de Competências a uma Rede Social Descentralizada para Utilização em Local de Trabalho

Monografia apresentada como requisito parcial
para conclusão do Curso de Computação — Licenciatura

Prof. Dr. Fernando William Cruz Prof. Dr. Wilson Henrique Veneziano
FGA/UnB CIC/UnB

Prof. Dr. Wilson Henrique Veneziano
Coordenador do Curso de Computação — Licenciatura

Brasília, 27 de dezembro de 2019

Dedicatória

Dedico a minha mãe, meu pai e minha irmã que me apoiaram. A Lorrana, que esteve ao meu lado nessa fase da minha vida. Aos meus amigos, que estiveram presentes em toda a minha jornada acadêmica.

Agradecimentos

Agradeço à Professora Doutora Germana pela dedicação, dicas, conversas e empenho que foram de grande ajuda e estímulo durante esse trabalho.

Resumo

O trabalho em equipe vem sendo cada vez mais adotado em ambiente corporativo como iniciativa institucional. Tanto em pequenas corporações, quanto naquelas geograficamente distribuídas, as Tecnologias da Informação e da Comunicação (TICs) têm um papel importante por várias razões, desde a capacidade de armazenamento e de cálculo dos computadores, até as possibilidades que se criam quando essas máquinas são dispostas em rede. Uma evidência são as chamadas Redes Sociais. Elas fazem agora parte do cotidiano do cidadão contemporâneo, e uma ideia promissora parece tirar proveito desse conceito em ambiente de trabalho. De fato, os clássicos sistemas para colaboração em ambiente corporativo, empreendidos pela comunidade ACM CSCW desde a década de 80, podem ser vistos como precursores da ideia. Por outro lado, a abordagem de gestão de competências tem atraído crescente interesse em organizações privadas e públicas, sobretudo quando respaldada pelas TICs. Buscando dar um primeiro passo para unir as duas abordagens, o trabalho ora proposto deve prover edição e consulta de competências aos membros de uma Rede Social ao integrar um editor de competências ao editor de perfil da Rede. O trabalho se insere no âmbito do projeto *Smart and Serious Social Networking in the Workplace* (engageGrid), que está por iniciar sua terceira fase e tem o objetivo mais amplo de prover apoio à formação de equipes em ambiente de trabalho.

Palavras-chave: Friendica, Rede Social, Rede Social Séria, Competência, Web Ontology Model, OWL, Local de Trabalho.

Abstract

Teamwork has been increasingly adopted in a corporate environment as an institutional initiative. Information and Communication Technologies (ICTs) play an important role in small and geographically distributed enterprises for a variety of reasons, from the storage and computing capacity of computers to the possibilities created when these machines are deployed network. One piece of evidence is called Social Networks. They are now part of the daily lives of the contemporary citizen, and a promising idea seems to be to take advantage of this concept in the workplace. In fact, the classic systems for collaboration in the corporate environment, undertaken by the ACM CSCW community since the 1980s, can be seen as the precursors of the idea. On the other hand, the competency management approach has attracted growing interest in private and public organizations, especially when it is being restructured by ICTs. Seeking to take a first step to unite the two approaches, the proposed work should provide editing and consultation of competency to the members of a Social Network by integrating a skills editor to the profile editor of the Network. The work falls within the ambit of the engageGrid project, which is about to begin its third phase and has the broader objective of providing support to the formation of teams in the work environment.

Keywords: Friendica, Social Network, Serious Social Network, Competency, Web Ontology Model, OWL, Workplace.

Sumário

1	Introdução	1
1.1	Contextualização: Projeto engageGrid e Gestão do Trabalho Colaborativo .	1
1.1.1	Breve Histórico	1
1.1.2	Concepção Atual	2
1.2	Motivação	3
1.3	Objetivos	3
1.4	Metodologia	4
1.5	Estrutura do Trabalho	4
2	Referencial Teórico	6
2.1	Redes Sociais	6
2.2	Linha do tempo	6
2.3	Rede Social, uma Selva	7
2.4	Redes Sociais em Ambiente de Trabalho	10
2.5	Friendica: um Ambiente de Código Aberto para uma Rede Social Descen- tralizada	11
2.5.1	Ótica Técnica	11
2.5.2	Ótica Acadêmica	12
2.5.3	Ótica do Usuário	13
2.6	Modelagem Computacional de Competências	16
2.6.1	O que Constitui a Competência?	16
2.6.2	Gestão de Competências	17
2.6.3	Modelagem Computacional	18
2.7	Persistindo Ontologias	20
2.7.1	Abordagens de Armazenamento	21
2.7.2	ARC2	21
2.7.3	Aprimoramento	22

2.8 Ambientes Computacionais mais Recentes que Exploram a Abordagem de Competências	23
2.8.1 Uma Extensão de Competências para o Moodle	23
2.8.2 TELOS	25
2.9 Considerações Finais Sobre o Capítulo	25
3 Proposta	26
3.1 Concepção de um Protótipo: Uma Primeira Versão	26
3.1.1 Desenvolvimento	26
3.1.2 Modelagem de Dados	29
3.1.3 Persistência em Base de Dados	29
3.1.4 Inserindo, Alterando, Excluindo (Ator: Usuário Owner)	31
3.1.5 Consultando (Ator: Gestor)	33
3.2 Análise de Capacidades e Limites	33
3.2.1 Do Editor	33
3.2.2 Do Modelo de Dados / Linguagem de Representação	33
4 Conclusão	35
4.1 Contribuições	35
4.2 Trabalhos Futuros	35
Referências	37
Anexo	39
I Instalação do Friendica	40
I.1 Instalação Friendica	40
I.1.1 Instalação do Xampp	40
I.1.2 Instalação do Friendica	41
I.2 Instalação Addon	46
II OWL	47

Lista de Figuras

1.1	Etapas do ciclo de vida e sua relação com a base de dados [1].	2
1.2	Representação do (fraco) acoplamento definido entre o conjunto de serviços da <i>Rede de Compromissos</i> e suas ferramentas para colaboração [2].	3
1.3	Interação entre o gestor do projeto e empregados com o serviço de competências.	4
2.1	<i>Redes Sociais</i> - Linha do tempo [3].	7
2.2	<i>Redes Sociais</i> dividido em sete blocos [3].	8
2.3	Menu de notificações e contador de mensagens privadas [4].	14
2.4	Visualização da linha do tempo, informações de contato exibidas, menu de controle aberto [4].	15
2.5	Calendário com <i>popup</i> de evento [4].	15
2.6	Postagem da comunidade exibida [4].	16
2.7	Integração: E-learning, Gestão de Competências em Recursos Humanos, Gestão do conhecimento e Business Process Management [5].	18
2.8	Gráfico representativo da Ontologia [5].	20
2.9	Gráfico representativo da Ontologia [5].	20
2.10	Gráfico representando a Ontologia utilizada com o Moodle em [6].	24
2.11	Tela de adição de competências [6].	24
3.1	Arquitetura aumentada.	29
3.2	Gráfico representativo da Ontologia.	30
3.3	Tabelas criadas pelo conversor de Banco de Dados do servidor ARC2.	30
3.4	Adição de competências.	31
3.5	Edição de competências.	32
3.6	Listagem de competências.	32
3.7	Consulta de competências.	33
I.1	Instalando Xampp.	41
I.2	Xampp Instalado.	42

I.3	Instalado Friendica terminal.	43
I.4	Instalado Friendica - Pré-requisitos.	44
I.5	Instalado Friendica - Banco de dados.	45
I.6	Instalado Friendica - Configuração.	45
I.7	Instalação concluída.	46

Capítulo 1

Introdução

Nos últimos anos as *Redes Sociais* sofreram constantes evoluções. A *Rede Social* chamada *Sixdegrees*, que surgiu em 1997, foi o primeiro sítio que permitia aos usuários criarem um perfil, adicionar, listar e adicionar contatos, dando a possibilidade de interação para que seus usuários pudessem compartilhar informações, como: fotos, textos, localização, etc[7]. Posteriormente surgiram outras *Redes Sociais*, como: *Linkedin*, *Orkut*, *Facebook*, *Twitter*, *Foursquare* e *YouTube*.

Uma das muitas *Rede Sociais* existentes é o *Friendica* [8] que conecta o usuário facilmente a uma *rede* de comunicação com alguns milhares de servidores. Atualmente, conta com mais de meio milhão de usuários registrados. Neste trabalho, ela foi escolhida para servir como base na integração da edição de competências. Os motivos da escolha estão descritos nos próximos capítulos.

1.1 Contextualização: Projeto engageGrid e Gestão do Trabalho Colaborativo

O projeto engageGrid - *Smart and Serious Social Networking in the Workplace* é um projeto com objetivo geral de abranger uma concepção, implementação e testes de um conjunto de serviços de software para manter uma *rede de compromissos* profissionais, provendo suporte computacional que permite uma comunidade assumir compromissos em grupo de maneira fundamentada e segura [1].

1.1.1 Breve Histórico

O projeto vem sendo elaborado desde 2008 e teve uma primeira etapa, no período de 2008-2010, que contou com o apoio do *Conselho Nacional de Desenvolvimento Científico e Tecnológico* (CNPq). Sua segunda etapa compreendeu o período 2012-2014.

Na primeira etapa, para concretizar o objetivo, foi desenvolvido um aparato conceitual, onde se definiu um ciclo de vida de um compromisso e houve sua prototipação a partir de um aparato tecnológico. Este aparato conta com a plataforma AGORA/GSD que serve de base para os serviços da *Rede de compromissos*. A vida de um compromisso no sistema, é dividida em uma estrutura com 3 bases de dados, que são: portfólio de competências, biblioteca de casos e perfil da comunidade. As bases de informações servem para alimentar uma primeira etapa, de cinco, do processo de gestão conhecido como ciclo de vida de um compromisso. Os cinco estágios do ciclo de vida de um compromisso são: configuração, negociação, realização, análise e síntese. A Figura 1.1 sintetiza a relação entre a base de dados e as etapas do ciclo de vida da *Rede compromissos* [1].

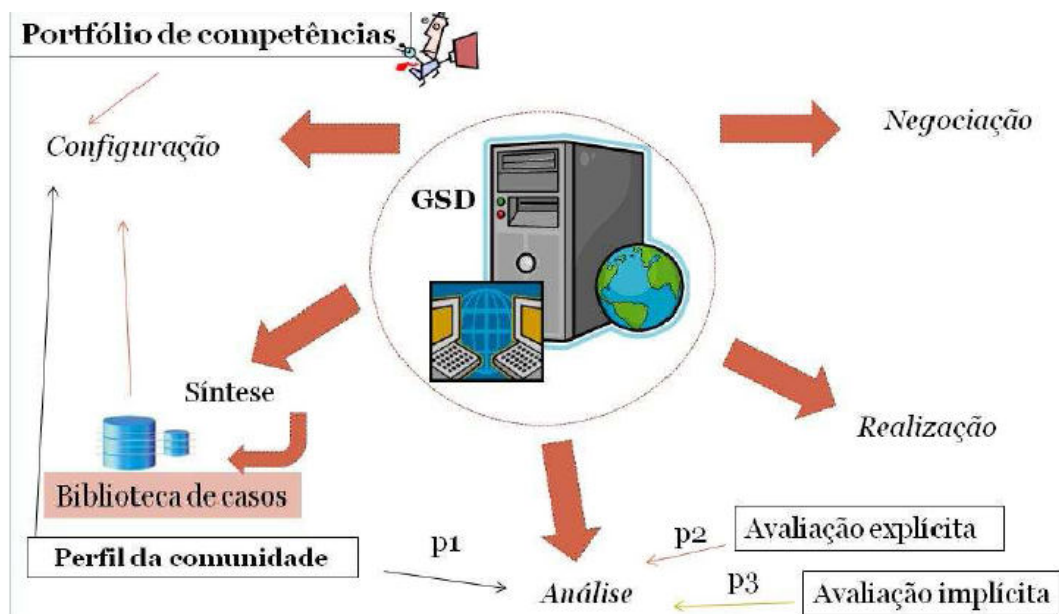


Figura 1.1: Etapas do ciclo de vida e sua relação com a base de dados [1].

1.1.2 Concepção Atual

Na primeira etapa do projeto, os protótipos foram concebidos de forma a ficarem embutidos na plataforma AGORA/GSD. A segunda etapa, teve foco em um fraco acoplamento entre os serviços da *Rede de Compromissos* e as ferramentas para prover colaboração [1].

Para isso o trabalho foi ramificado em três outros projetos. O primeiro abordava os serviços da *Rede de Compromissos*. O segundo foi orientado para a criação e a manutenção de um nó *Grid Shared Desktop* (GSD) específico. E o terceiro trabalho versou sobre a inclusão de uma ferramenta popular como demonstrador utilizando o GoogleDocs como sua base [2].

A concepção atual é demonstrada na Figura 1.2.

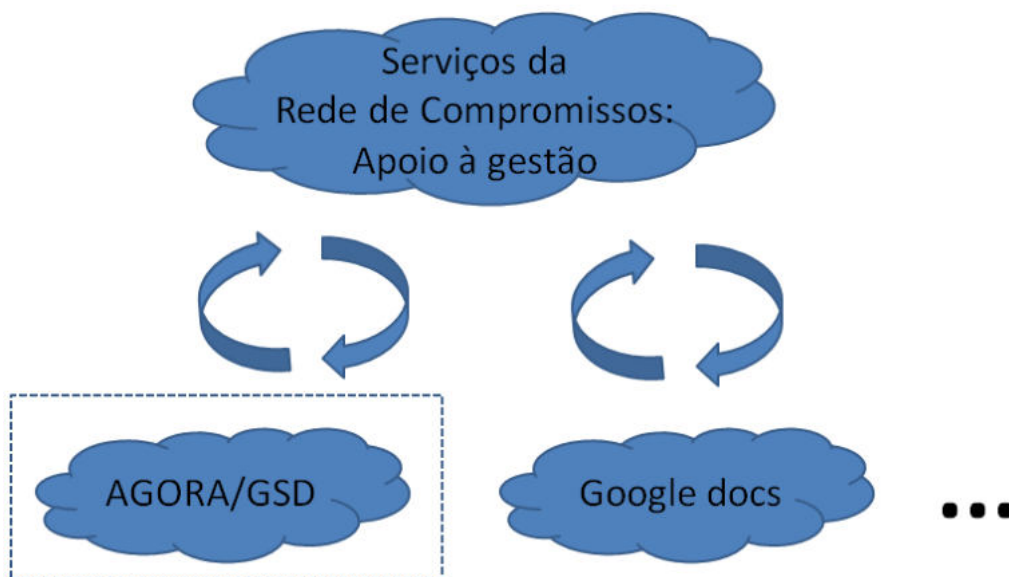


Figura 1.2: Representação do (fraco) acoplamento definido entre o conjunto de serviços da *Rede de Compromissos* e suas ferramentas para colaboração [2].

1.2 Motivação

As *Redes Sociais* estão cada vez mais presentes no cotidiano do cidadão contemporâneo, empregadas em ambientes de trabalho em *Redes Sociais Sérias*. Com o aumento da oscilação dos funcionários em ambientes corporativos, consequentemente, a evolução de recursos humanos enfrenta dificuldades ao tentar manter o ambiente em rápida mudança[2].

1.3 Objetivos

O objetivo deste trabalho é o estudo de viabilidade entre uma *Rede Social* e um sistema de gestão de competências. Ela segue a ideia da Figura 1.3, onde os serviços da **rede**, que funcionam em nuvem, fornece serviços para os gestores e colaboradores.

Os colaboradores, por sua vez, cadastram suas competências enquanto os gestores podem pesquisar qual empregado tem a competência que se enquadra nos requisitos do projeto a ser executado.

Para isso, este trabalho deve prover a edição e consulta de competências aos membros de uma *Rede Social*, integrando um editor de competências ao editor de perfil da *rede*.

Para os serviços da *Rede de Compromissos* foi adotado a *Rede Social Friendica* que hoje está incorporada no cotidiano do cidadão, onde sua contribuição vai além da questão da interface.

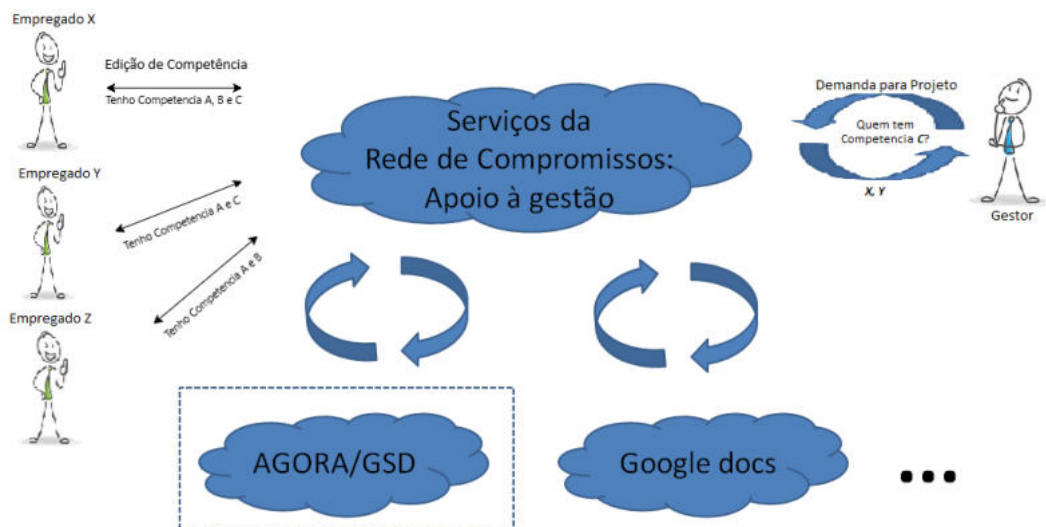


Figura 1.3: Interação entre o gestor do projeto e empregados com o serviço de competências.

O objetivo geral deste trabalho é prover a integração entre o *Friendica* e um sistema de gestão de competências. Para atingir o objetivo geral, foi definido o desenvolvimento de um *Addon* para o *Friendica*, que forneça o serviço.

Os objetivos específicos são os seguintes:

1. Adicionar, editar e remover competências;
2. Prover mecanismo de busca das competências;

1.4 Metodologia

A natureza deste trabalho pode ser classificada como aplicada. O *Friendica* e os modelos de competências são estudados com a finalidade de contribuir para o estudo da viabilidade da integração propostas nos objetivos.

A literatura foi explorada para pesquisar e selecionar *Redes Sociais* e modelos de competência para a integração.

1.5 Estrutura do Trabalho

Esta monografia é composta por quatro capítulos, incluindo esta introdução.

No capítulo 2, é feita uma descrição de todo o referencial teórico utilizado como base no desenvolvimento deste trabalho, incluindo os aspectos técnicos e conceituais acerca de competências, análise das *redes sociais* em ambiente de trabalho e uma apresentação

do *Friendica* com ótica acadêmica, técnica e do usuário. Aborda também a gestão e modelagem computacional de competências.

O capítulo 3 trata dos aspectos técnicos e conceituais do protótipo desenvolvido, apresentando a concepção do protótipo, a modelagem de dados e a persistência na Base de Dados.

Por último, no capítulo 4, encerra-se o trabalho, apresentando conclusões, contribuições e projetos futuros.

Em anexo, é apresentado um tutorial de instalação do *Friendica* e do *Addon* em ambiente *Linux* e do modelo que foi utilizado para competências.

Capítulo 2

Referencial Teórico

Este capítulo apresenta os elementos base para o trabalho desenvolvido nesta monografia.

2.1 Redes Sociais

Os autores Boyd & Ellison [9] definem o sítio de *Rede Social* como um serviço baseado na Web que permite aos usuários criarem um perfil público ou semi-público dentro de um sistema delimitado. Esses indivíduos podem gerir sua própria lista de amigos e ter acesso a outros usuários com os quais compartilham uma conexão e aquelas feitas por outras pessoas dentro do sistema. A natureza e a nomenclatura dessas conexões podem variar de sítio para sítio. Um dos primeiros sítios que permitia ao usuário criar um perfil, listar seus amigos e adicionar amigos de amigos para sua própria lista, surgiu em 1997 com o sítio de *Rede Social* chamado *Sixdegrees*.

2.2 Linha do tempo

Na Figura 2.1, mostramos uma cronologia do surgimento de algumas *Redes Sociais*.

A pioneira foi *Sixdegrees*, surgindo no ano de 1997. Em 1999 a LiveJournal, seguida por AsianAvenue e BlackPlanet. No ano 2000 a MiGente e. Em 2001 surgiram Cyworld e Ryze e em 2002: Fotolog, Friendster e Skyblog. Em janeiro 2004 surge o Orkut e seu fechamento se dá em 30 de setembro de 2014. Nos anos seguintes entre as muitas *Redes Sociais* surgentes de propriedade privada, se destacam: *Facebook*, que atualmente conta com mais de dois bilhões de usuários, *Twitter*, *LinkedIn*, *Foursquare* e *YouTube*. As de código aberto, temos: *Mastodon*, *Diaspora*, *GnuSocial*, *Pleroma*, *Hubzilla* e *Friendica*.

Com o crescimento dos sítios de *Redes Sociais*, milhões de usuários estão compartilhando informações com amigos e seguidores, isso compartilhamento de dados em tempo

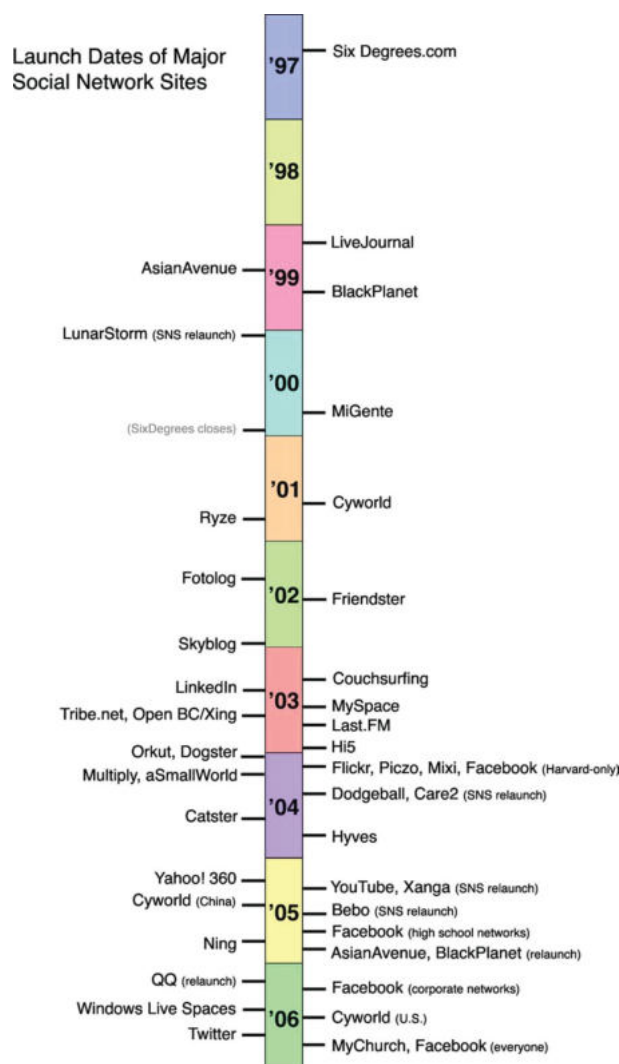


Figura 2.1: *Redes Sociais* - Linha do tempo [3].

real. Estes dados variam de notícias pessoais (como atualizações de *status*) a notícias globais (como notícias sobre política, ciência, esportes, tecnologias, etc.).

2.3 Rede Social, uma Selva

No artigo "Social media? Get serious! *Understanding the functional building blocks of social media*"[3], os autores vão um pouco mais longe que Boyd & Ellison [9] e fazem uma analogia da *Rede Social* com uma selva e seus ecossistemas. Assim, dividem a *Rede Social* em sete blocos, são eles: identidade, conversas, compartilhamento, presença, relacionamentos, reputação e grupos. Mostramos esses blocos na Figura 2.2.

As *Redes Sociais* não se limitam em ter apenas um bloco em suas funcionalidades, apesar que, pode ser conveniente existir um bloco com mais evidência. Normalmente são

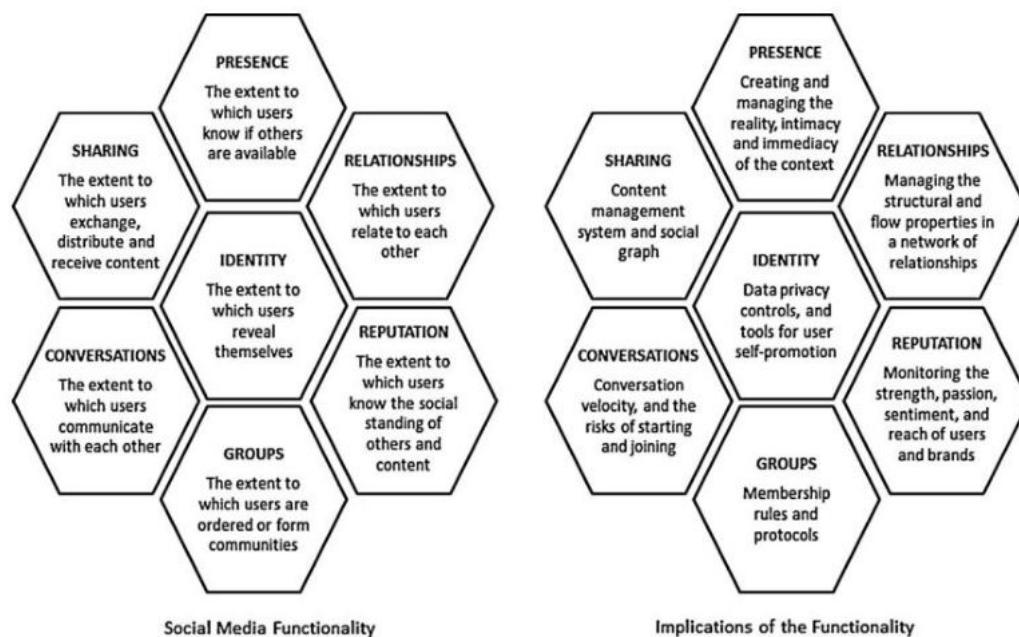


Figura 2.2: *Redes Sociais* dividido em sete blocos [3].

compostas por três ou quatro blocos, variando de acordo com sua natureza. É possível classificar as *Rede Sociais* e os blocos:

1. *Facebook*, que tem o bloco de relacionamentos como o seu maior motor, e em menor destaque as conversas;
2. No caso do *LinkedIn*, o seu bloco mais forte é a identidade. Nele encontramos também a reputação, onde cada individuo menciona as suas qualidades e outras pessoas pertencentes ao seu grupo de amigos atestam a veracidade;
3. *YouTube* como um exemplo de compartilhamento de vídeos.
4. *Foursquare* com o seu maior bloco sendo a presença, na qual seus usuários compartilham onde estão e suas opiniões de tal lugar;
5. No Twitter temos o bloco de conversas;

O *LinkedIn* é uma *Rede Social* para profissionais apresentarem suas competências. É utilizada principalmente para estabelecer relações comerciais e procurar um novo emprego. Foi fundado em 2003 e atualmente a *Rede* está presente em mais de duzentos países e disponível em pelo menos dez idiomas. Tem como objetivo que os membros registrados possam estabelecer conexão com outras pessoas. Esses usuários têm a possibilidade de convidar outros usuários para fazer parte da sua *Rede* de conexão. Quanto maior a quantidade de conexões com outros usuários, maior sua rede, aumentando o contanto com outros profissionais.

Os usuários podem cadastrar o seu currículo, dando ênfase em suas competências. Essas competências podem ser atestadas por outros usuários que estão em seu ciclo de amigos.

Atualmente o uso de *Redes Sociais*, como o *LinkedIn*, está sendo cada vez mais utilizada para selecionar e recrutar funcionários. As empresas também usam o *Facebook*, *YouTube* e o *Twitter* como uma ferramenta para estabelecer um relacionamento mais próximo com o público e atrair novos clientes [10].

2.4 Redes Sociais em Ambiente de Trabalho

Muitas companhias usam as *Redes Sociais* como uma importante ferramenta para comunicação com o seu público alvo e seus clientes. As *Redes Sociais* se tornaram um poderoso e rápido meio de comunicação, pois em segundos um novo produto pode ser visualizado e conhecido por dezenas ou até milhares de pessoas.

Em um de seus trabalhos, David Millen [11], expõe como a inclusão de uma rede social dentro de uma grande empresa permite um novo método de comunicação entre os colegas, trazendo tanto o compartilhamento pessoal quanto o profissional dentro das paredes protegidas de uma intranet.

Em uma de suas palestras feita no Simpósio Brasileiro de Sistemas Colaborativos em 2010, David Millen já apresentava os seus estudos sobre o crescimento rápido e generalizado de software social em ambientes organizacionais e de trabalho, assim também como os novos desafios e oportunidades que com elas surgiram.

Robert [12] aborda os três estágios do uso da tecnologia social identificados pela Professora Harquail do *Stevens Institute of Technology*, são eles:

1. O uso da *Rede Social* em um compromisso focado e limitado entre funcionários.
2. O uso da tecnologia, *Rede Social*, que atravessa a barreira interna, como o gerenciamento do relacionamento focado na relação com o cliente.
3. Incorporar a *Rede Social* na cultura do local de trabalho, transformando a forma como os funcionários pensam sobre o que fazem e como trabalham juntos.

Algumas empresas ainda possuem a atitude de bloquear o uso de *Redes Sociais* em suas dependências, não toma partido a importância e os benefícios de ter em sua equipe pessoas experientes no assunto.

Em contrapartida, compreender e ter a estratégia certa pode ajudar no uso inteligente das *Redes Sociais*, como abordado por Ferris [13], onde empresas que planejaram um método para o uso da *Rede Social* visando aumentar a coesão da equipe conseguiram alcançar lucros e até mesmo fazer economias. Temos, como exemplo, o Departamento de Trabalho e Pensões do Reino Unido que economizou cerca de 21 milhões de libras com um jogo, em sua *Rede Social* interna, onde os funcionários sugerirão melhorias, onde as melhores sugestões foram implementadas através de abordagem em equipe.

Um dos primeiros exemplos que mostra a rápida comunicação das *Redes Sociais*, ocorreu em julho de 2009, quando a empresa *United Airlines* de transporte aéreo, avariou o violão do cantor Dave Carroll. O cantor e sua banda estavam viajando da cidade canadense de Halifax rumo ao estado norte-americano de Nebraska. Ao chegar em seu destino, constatou que seu violão foi danificado. Após tentativas mal sucedidas de receber uma

indenização, ele compôs uma música e fez um vídeo clipe, publicando no *YouTube* com nome de *United Breaks Guitars*, relatando sua experiência com a empresa. O vídeo foi viral e obteve, em pouco tempo, nove milhões e meio de visualizações [3].

A empresa *United Airlines*, depois de alguns dias da publicação do vídeo, entrou em contato com o cantor, como ele mesmo explica em outro vídeo, também publicado no *YouTube* [14].

Após a repercussão, outras pessoas manifestaram a insatisfação com a empresa. Atualmente, o vídeo clipe original permanece no *YouTube* [15] com pouco mais de dezenove milhões visualizações.

2.5 Friendica: um Ambiente de Código Aberto para uma Rede Social Descentralizada

2.5.1 Ótica Técnica

Alguns pontos positivos, dos quais fazem parte para a escolha da utilização do *Friendica* neste trabalho, são: tem uma ênfase em opções avançadas de privacidade; tem uma fácil instalação; controlar facilmente os dados pessoais; objetiva a comunicação com outras *Redes Sociais* quanto forem possíveis; é descentralizada, os relacionamentos podem ser feitos em qualquer sistema compatível; é possível criar uma rede de escala de Internet composta de sítios menores; tem um enfoque em privacidade, tendo lista de acesso para cada item; grupos de conversas particulares, onde todas as comunicações são restritas aos membros do grupo; mensagens privadas um-para-um em protocolos suportados; é possível expirar conteúdo antigo após um determinado período de tempo; os dados dos usuários não é pertencente a segundos nem terceiros, ou seja, todos os dados pessoais são verdadeiramente pessoais [8].

Generalidades

Com o *Friendica*, é possível interagir totalmente com qualquer pessoa no *Twitter*, postar no *Facebook* e receber qualquer conteúdo no *Tumblr*, *Wordpress* ou *RSS*. Permite a integração com a maioria dos sítios através de uma variedade de *Addons*.

Os requisitos para instalação do *Friendica* são: *Apache*, *PHP* e *MySQL* ou uma alternativa equivalente é o *XAMPP* [16] que prove tais funcionalidades. Um tutorial completo da instalação utilizando o *XAMPP* em ambiente *Linux* (*Ubuntu*) está disponível em anexo.

Addons

Existe uma coleção de *Addons* para o *Friendica* que é possível adicionar ao diretório addon da instalação. Entre os *Addons*, existem os conectores para outras *Redes Sociais*, como por exemplo: *Twitter*, *Pump.io* e *Google+*. Estes conectores são necessários quando o protocolo não é suportado pelo núcleo do *Friendica*.

Para ativar os *Addons*, conectores ou as conexões para outras *redes* suportadas pelo *Friendica* (*GNU social*, *Diaspora* e *Red Matrix*, entre outras), basta acessar a configuração da página no painel do administrador e ativá-los [17].

A interface *Addon* do *Friendica* é muito flexível e poderosa, com isso, é possível adicionar quase todos os tipos de funcionalidades. Também é possível adicionar traduções e multi idiomas, utilizando o projeto *Transifex* para o *Friendica*.

O nome do *Addon* não pode conter espaços ou outras pontuações. Este nome é usado para os arquivos e funções internas. Cada *Addon* deve conter uma função de instalação e de desinstalação. Essas duas funções não recebem argumentos e serão chamadas se o complemento for alterado após a instalação (ou seja, reinstalado). Portanto, sua desinstalação não deve destruir os dados e a instalação deve considerar que os dados já podem existir [17].

O *Addon* devm conter um bloco de comentários com os quatro parâmetros apresentados a seguir:

```
1 / *
2 * Name: Friendica competency
3 * Description: Friendica crud competences.
4 * Version: 1.0
5 * Author: Norberto Luz Caires Junior <beto_caires@hotmail.com>
6 * /
```

É necessário também adicionar um arquivo *README* ou *README.md* ao diretório do *Addon*. Ele será exibido no painel de administração e deve incluir algumas informações adicionais.

2.5.2 Ótica Acadêmica

Em seu trabalho de 2015, Vassileva [18] constrói um *plugin* para o *Friendica*, nomeado como MADMICA 1. Esse *plugin* tem como objetivo utilizar um método interativo para que os usuários visualizem e personalizem a filtragem dos conteúdos das publicações que serão apresentados em sua página na *Rede Social*. Os filtros atualmente existentes conduzem frequentemente ao problema da *bolha de filtro*, que é um termo referente a usuários que são encapsulados em fluxos de dados, como notícias ou atualizações de *Redes Sociais*

não personalizadas para seus interesses. Ou seja, os usuários só podem receber informações que corresponda suas preferências pré-estabelecidas, mas não são expostas a novos tópicos, isso acontece devido ao foco de precisão. Além disso, a maioria dos códigos que fazem essa filtragem são caixas pretas, não dando nenhuma transparência para o usuário sobre como o mecanismo de filtragem decide o que deve ser apresentado no fluxo de atividades.

A escolha do *Friendica* para o uso no *MADMICA 1*, se dá por:

1. Código aberto;
2. Tem interface federada, ou seja, objetivando a comunicação com outras *Redes Sociais* quando forem possíveis;
3. Descentralização. É possível criar uma rede de escala de Internet composta de sítios menores, e o usuário escolhe onde seus dados serão alocados.

2.5.3 Ótica do Usuário

Uma serie de imagens são apresentadas nesta seção para demonstrar a interface visual a que um usuário tem acesso.

A Figura 2.3 mostra como é a interface do menu de notificações e o contador de mensagens privadas.

Na Figura 2.4 temos a visualização da linha do tempo, com as informações de contato exibidas e o menu de controle aberto.

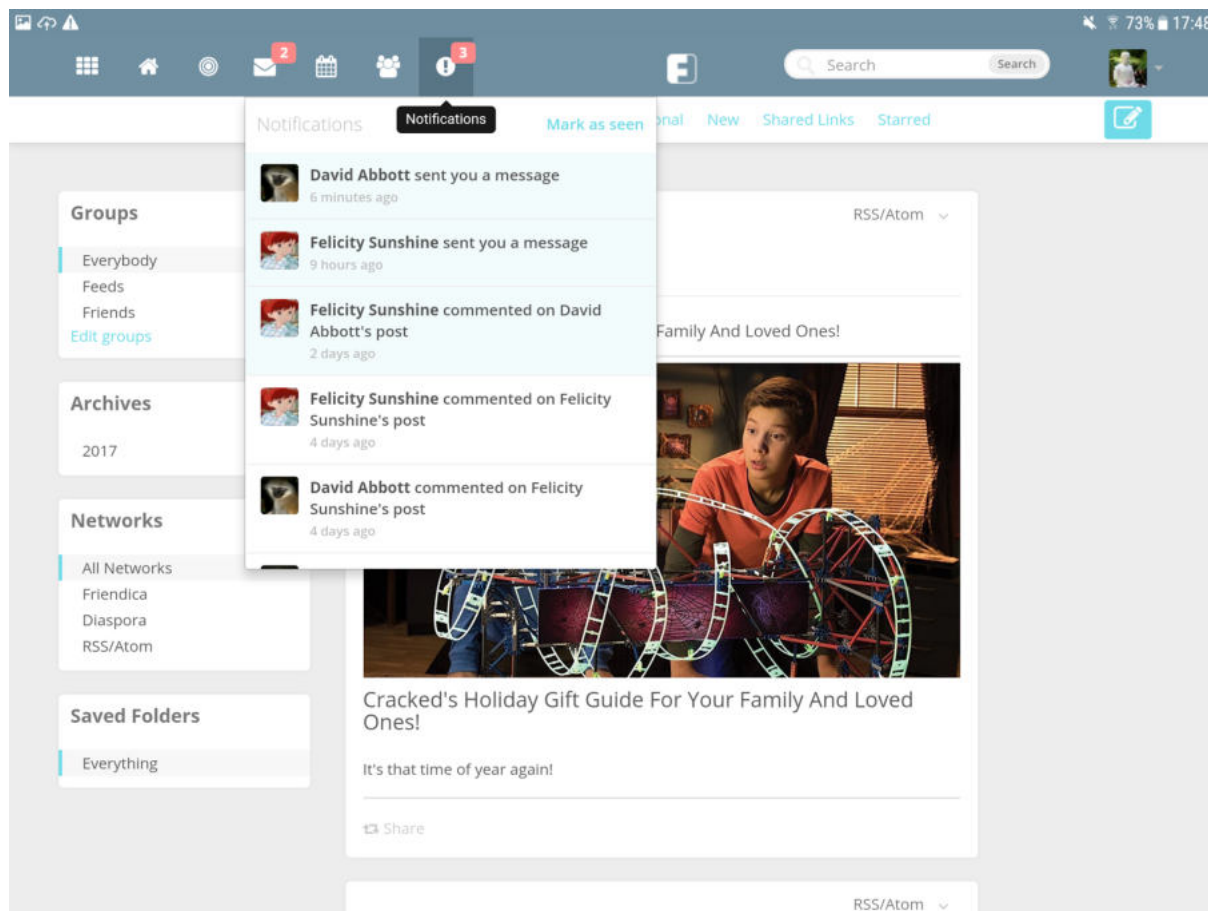


Figura 2.3: Menu de notificações e contador de mensagens privadas [4].

Na Figura 2.5 é possível visualizar o calendário de notificações, que é como uma agenda, porém digital onde tem um *popup* para notificar os eventos.

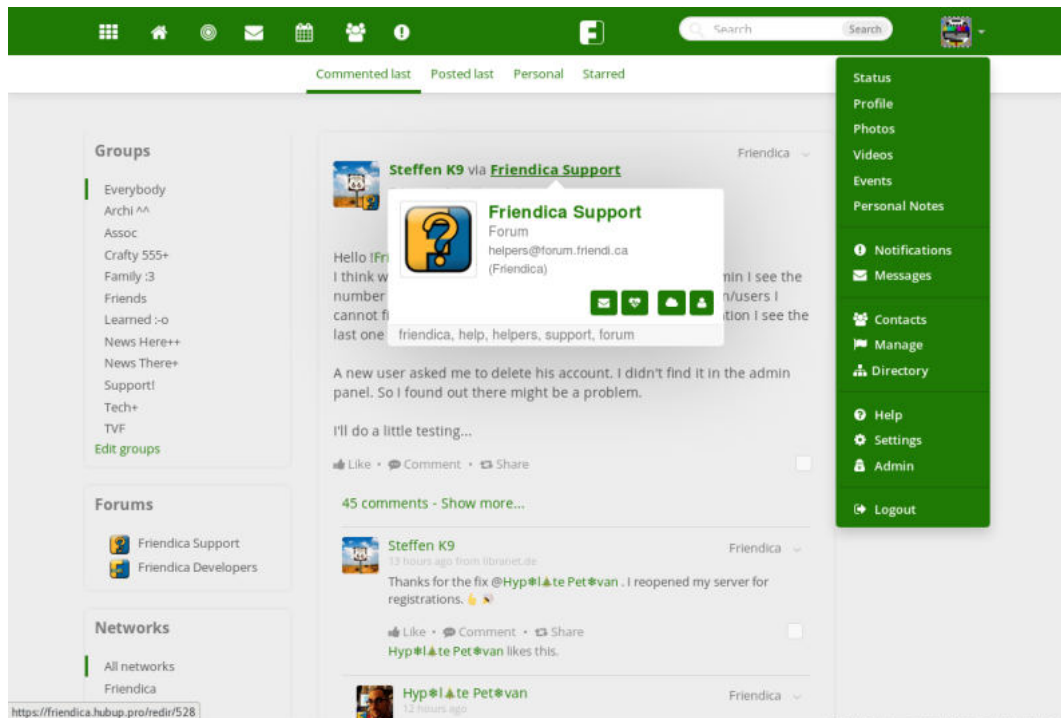


Figura 2.4: Visualização da linha do tempo, informações de contato exibidas, menu de controle aberto [4].

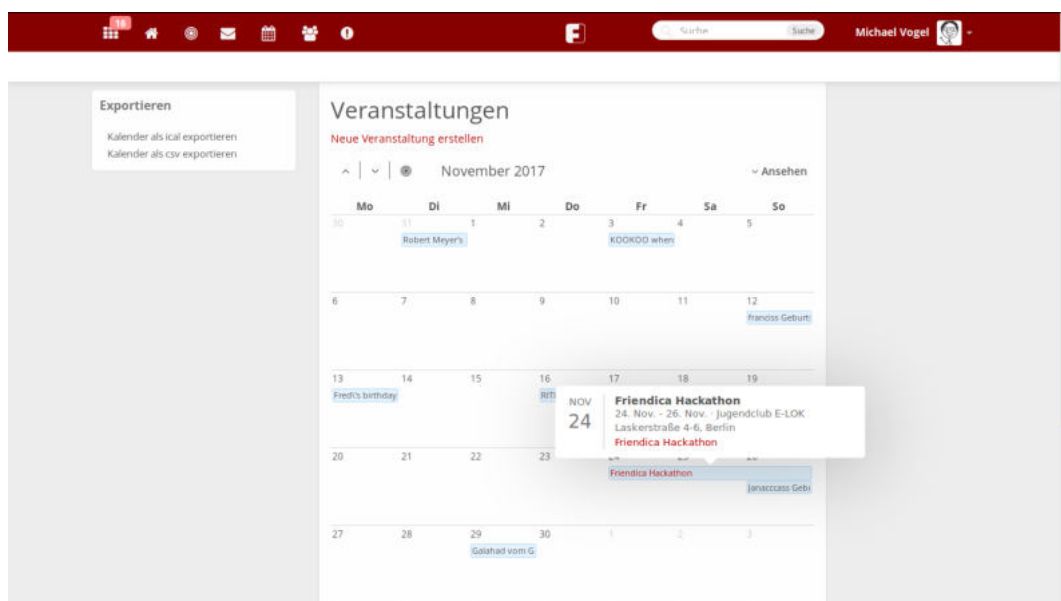


Figura 2.5: Calendário com *popup* de evento [4].

A Figura 2.6 mostra uma postagem em uma comunidade do *Friendica*.

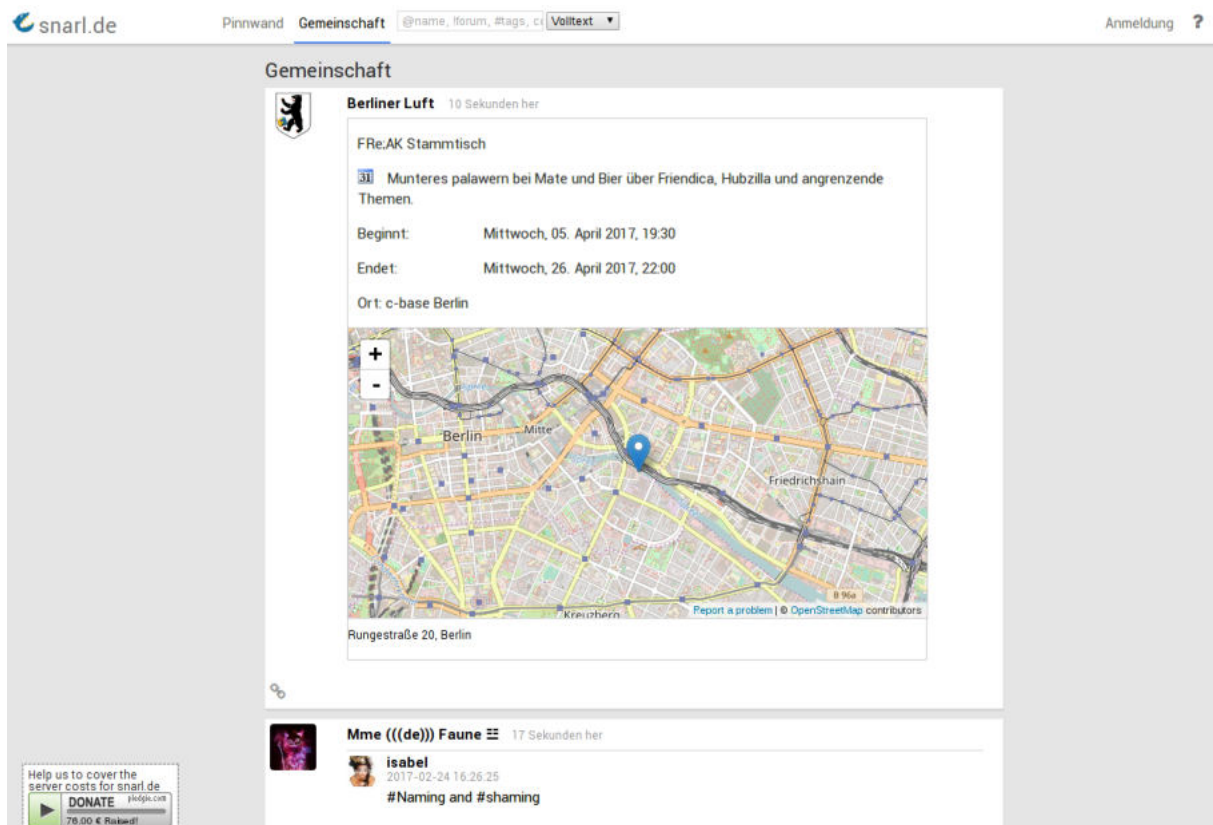


Figura 2.6: Postagem da comunidade exibida [4].

2.6 Modelagem Computacional de Competências

Esta seção traz elementos que são necessários para a compreensão de Competências e da Modelagem Computacional de Competências. São apresentadas *definições* e *modelos computacionais*.

2.6.1 O que Constitui a Aompetência?

A definição de competência é uma questão amplamente debatida. O escopo deste trabalho restringe-se a fornecer a definição usada no projeto engageGrid.

O termo competência é derivada da palavra latina *competere* e utilizando sua tradução o seu significado é: *ser adequado*. Está é uma definição da palavra em si e diversos outros autores citam diferentes *definições* dependendo do contexto de estudo. Por exemplo, o conceito de competência tem origem na Psicologia, que explora a eficácia com que uma pessoa se comporta com problemas que se apresentam em seu ambiente [19].

Os pesquisadores tem uma discordância no significado entre as palavras *competence* e *competency*. Alguns escritores usam a palavra *competencies* como plural para *competence* ou tratam elas como sinônimos. Mas, como Cheetham e Chivers [20] argumentam,

competency é um subconjunto das *competencies* para alguns campos profissionais ou acadêmicos.

Com base no trabalho dos autores Cheetham e Chivers [21], uma competência é definida como o desempenho efetivo de um agente em um ambiente bem definido. Também observam que uma competência é: situacional; um atributo latente identificada e definida em uma comunidade de prática; uma disposição de um indivíduo que pode ser atribuída não somente a ele, mas também para equipes e organizações.

Para desempenhar trabalhos em um cenário característico, um indivíduo deve apresentar ou adquirir competências que podem ser definidas como um agrupamento de habilidades, saberes, comportamentos, onde o seu grau de desempenho varia do mais baixo nível de proficiência para os mais elevados níveis de perfeição.

Na seções a seguir, e com base na definição dada nos parágrafos anteriores, apresentamos a questão da gestão e modelagem computacional de competências.

2.6.2 Gestão de Competências

Tobias [22] define gestão de competências como “abranger todos os instrumentos e métodos usados em uma organização, para sistematizar competências atuais e futuras necessárias, para que o trabalho seja realizado e avaliar as competências disponíveis da força de trabalho”. Hamel e Prahalad [23], ainda em 1994, antecipou que “com o passar do tempo, a gestão de competência será cada vez mais importante: a competência será bem reconhecida como extremamente notável para o alcance dos objetivos da empresa e complementar, por exemplo, aos principais processos de negócios, relações com os clientes, questões financeiras e assim por diante”.

Schmidt [5] defende que a capacidade dos funcionários é de suma importância para as empresas. Suas competências e habilidades definem: satisfação do público alvo, a eficiência e a criatividade. Com o desenvolvimento acelerado do segmento do comércio e o aumento da oscilação dos funcionários, a evolução de recursos humanos enfrenta dificuldades ao tentar manter o ambiente em rápida mudança. Com os espaços dinâmicos, os meios tradicionais de aperfeiçoamento de capacitação sofrem com suas limitações. Uma alteração no modelo em direção à aprendizagem sob demanda tem que ser feita. Porém, necessita um engajamento de processos de ensino e processos de trabalho. Com o aprofundamento da integração destes dois, torna-se óbvio que a aprendizagem é uma preocupação transversal de vários processos corporativos, entre os quais os mais importantes são:

1. E-learning (como a abordagem à aprendizagem individual apoiada por tecnologia e apoiada pedagogicamente)

2. Gestão de Competências em Recursos Humanos (como o alinhamento da aprendizagem individual com estratégias corporativas)
3. Gestão do conhecimento (como a perspectiva organizacional sobre a aprendizagem dos funcionários)
4. Business Process Management (com o processo de negócios sendo o quadro e o contexto primário de todas as atividades relacionadas à aprendizagem nas empresas)

A Figura 2.7 mostra essas integrações.

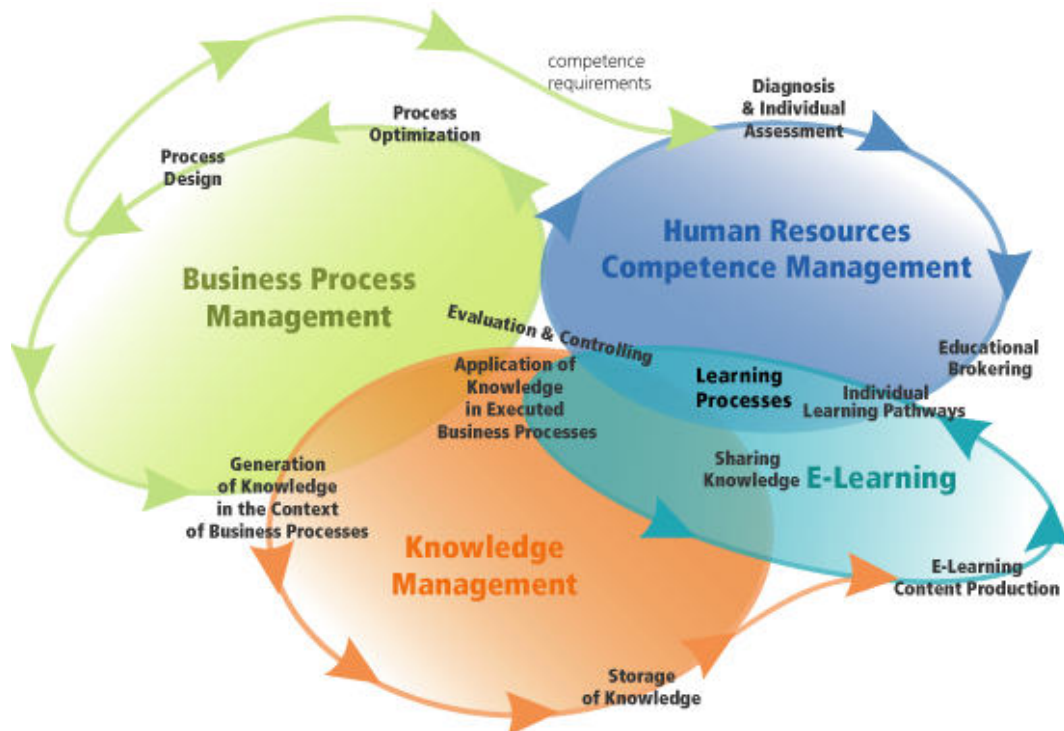


Figura 2.7: Integração: E-learning, Gestão de Competências em Recursos Humanos, Gestão do conhecimento e Business Process Management [5].

A dificuldade na associação desses modelos é que as inúmeras disciplinas criam recursos mais ou menos isolados, tendo diferentes conceitos subjacentes de aprendizado e conhecimento. Superar as desigualdades e conceitos foi o ponto inicial da pesquisa integrativa.

2.6.3 Modelagem Computacional

Existem algumas especificações para descrição de competência, como: o IMS RDCEO, o IEEE RCD, o HR-XML e o OWL. A criação dessas especificações tem por objetivo facilitar a troca de dados entre sistemas que lidam com informações relacionadas à competência, permitindo consultar *definições* comuns com valores comumente reconhecidos.

O RDCEO – *Reusable Definition of Competency or Educational Objective* do consórcio IMS (IMS RDCEO, 2002) tem o objetivo de definir uma informação para descrever, referenciar e trocar *definições* de competências e objetivos educacionais, ampliando sua tecnologia para o treinamento fundamentado em competência.

Os elementos que compõem o RDCEO são: três obrigatórios (identificador global único, título e descrição) e dois opcionais, que são eles: definição e metadados.

Citando o autor Sampson [24]: "Embora a especificação IMS RDCEO não pretende oferecer uma solução para a agregação de competências complexas a partir de competências simples, seu modelo de dados permite a integração de informações internacionais ou ontologias de competência através da incorporação de metadados. Além disso, a especificação IMS RDCEO não aborda a questão de como as competências devem ser avaliadas, certificadas, registradas ou usado como parte de um processo com um escopo mais amplo, como o Gerenciamento de Competências."

A especificação IEEE RCD (*Reusable Competency Definitions*) descreve uma definição de Competência, prevista para ser usada em um Sistema de Gerenciamento de Aprendizagem ou referenciado em um Perfil de Competência, fazendo referência direta do IMS Especificação RDCEO [24].

Os elementos que compõem o IEEE RCD são: dois obrigatórios (identificador global único e título) e três opcionais, que são eles: descrição, definição e metadados.

O Consórcio HR-XML [24] é fundamentado em estabelecer uma definição da competência, podendo registrar evidências aplicadas em substanciar, podendo dar classificações e pesos para serem usadas para relacionar, verificar e qualificar as habilidades ou interesses de uma competência.

A HR-XML ocupa-se de como medir a competência de funcionários em uma empresa utilizando níveis de proficiência. Também desse padrão veio a base para o uso de evidências de aprendizagem nos cálculos do desempenho dos aprendizes, bem como a instituição de peso para valorar como essas evidências influenciam o desempenho, qualificação e interesse influenciam na competência de uma pessoa.

A Web Ontology Language (OWL) [25] foi concebida para organizar uma linguagem de ontologia que pudesse ser usada para representar as classes e relacionamentos entre elas em documentos e aplicações Web. Os elementos de uma ontologia precisam ser referenciados de modo que possibilite expor sem equívoco e aplicados por agentes de software.

Nas Figuras 2.8 e 2.9, mostramos o grafo representando a Ontologia em OWL de competência que será utilizada neste trabalho.

O modelo completo é mostrado no anexo II.

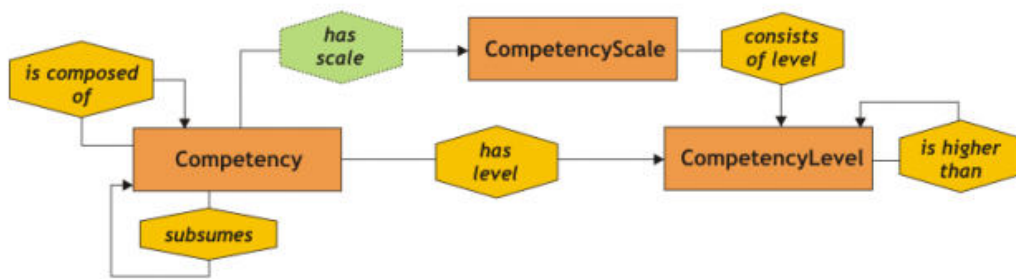


Figura 2.8: Gráfico representativo da Ontologia [5].

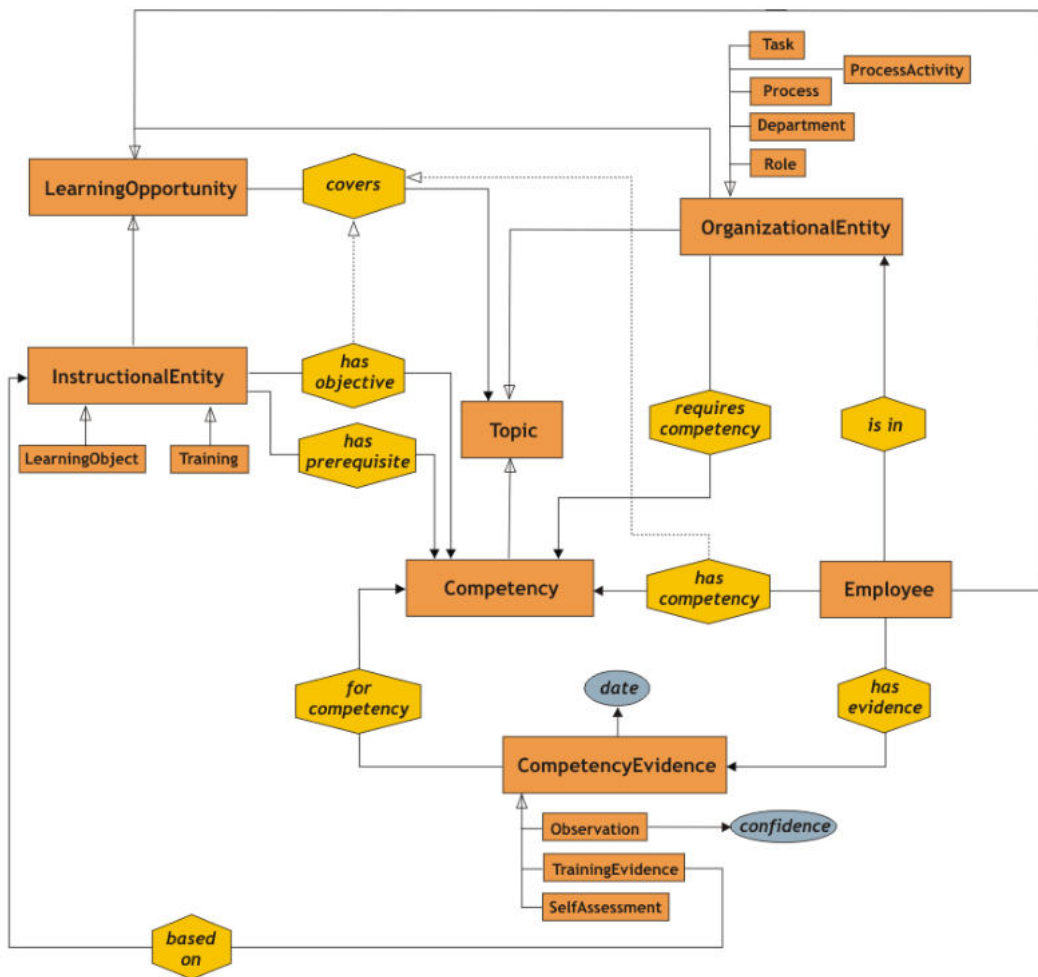


Figura 2.9: Gráfico representativo da Ontologia [5].

2.7 Persistindo Ontologias

Esta seção trata sobre paradigmas e tecnologias para a persistência de ontologias.

2.7.1 Abordagens de Armazenamento

Existem várias abordagens para armazenamento de dados RDF¹/OWL, abaixo identificam-se três abordagens com base na técnica [26] que usam para a persistência:

1. Armazenamento em memória. Neste método aloca-se uma quantidade de memória principal que ficará disponível para armazenar os dados RDF/OWL. Ponto negativo é que a memória é limitada e escassa, pequena quantidade de dados RDF podem ser armazenados.
2. Armazenamento do sistema de arquivos. Além de ser uma maneira de armazenar dados OWL permanentemente, tem uma fácil implementação e simplicidade de armazenar em um arquivo único. O problema é que requer o gerenciamento manual dos arquivos OWL e não são adequados para armazenar, carregar e pesquisar com eficiência uma grande quantidade, pois gera uma insuficiência na gestão e desempenho.
3. Banco de dados relacional. Esta abordagem aplica sistemas de bancos de dados relacionais para armazenar e gerenciar dados RDF/OWL permanentemente.

Atualmente, existem vários repositórios semânticos disponíveis como projetos de código aberto e soluções proprietárias: Sesame, ARC2, Jena, Kowari Metastore, Allegro-Graph e Virtuoso.

Para este projeto, foi utilizado o ARC2 que será apresentado na seção a seguir.

2.7.2 ARC2

O ARC é distribuído gratuitamente sob a licença de software *W3C*®. Começou a ser desenvolvido em 2004 como um sistema RDF leve para analisar e serializar arquivos RDF/XML. Mais tarde evoluiu para um framework mais completo com funcionalidade de armazenamento e consulta.

O ARC é escrito em PHP, com isso facilita a integração de RDF e SPARQL em estruturas PHP/MySQL. O seu desenvolvimento foi direcionado para projetos Web convencionais e foi otimizado para ambientes da Web compartilhados (ou com desempenho limitado ou com privilégios limitados). Em 2011, se tornou uma das bibliotecas RDF mais utilizadas. A sua documentação, além de extensa é clara e concisa, sendo um dos fatores importantes para o seu sucesso. O ARC2 surgiu com a atualização do ARC em janeiro de 2014 [27].

¹O RDF (*Resource Description Framework*) é uma estrutura para descrever recursos na Web que são escritos em XML, foi projetado para ser lido e entendido por computadores, não sendo projetado para ser exibido para pessoas.

O suporte ao SPARQL se dá por meio de consultas do tipo: SELECT, CONSTRUCT, ASK e DESCRIBE, incluindo OPTIONALs, UNIONs simples e filtros REGEX nas variáveis. Os métodos podem ser chamados através de uma interface de programação. Os formatos dos resultados das consultas podem ser: JSON (+JSONP/JSONI), XML, COUNT, RDF/XML e Turtle/N3 [28].

Ele faz com que o PHP processe o mínimo possível, fazendo com que quase todo o processamento fique a cargo do Banco de Dados. Embora use principalmente código orientado a objetos, os dados processados são mantidos como objetos PHP nativos, ou seja, usa matrizes de baixa memória ao invés de instâncias de objetos personalizados. O impacto do uso do PHP não é muito grande, a escalabilidade do *ARC* pode ser aumentada em grande parte ajustando o Banco de Dados e vários parâmetros de configuração, por exemplo *buffers* para carregamento em massa [28].

O *ARC* não depende de variáveis globais, constantes ou outras práticas que geralmente causam problemas de integração. As funções e componentes são tão pequenas quanto possível, apenas o código imediatamente requerido é carregado na memória.

O seu desenvolvimento continua, as fontes estão disponíveis para a comunidade por meio do GitHub [28].

2.7.3 Aprimoramento

Grande parte dos repositórios Web semânticos requer o gerenciamento de grandes quantidades de dados. O gerenciamento eficiente dos modelos semânticos é um fator crítico na determinação do desempenho de muitos aplicativos. Isto foi a motivação para a criação de repositórios especializados.

Em seu artigo Almari [26] propõem um layout aprimorado para o banco de dados relacional para permitir o armazenamento da ontologia OWL de maneira efetiva. O esboço da solução é propor um novo modelo de armazenamento de OWL adotando várias alterações na forma de gestão dos dados, a fim de projetar um modelo de banco de dados relacional eficiente para armazenar bases de conhecimento de ontologias.

Esta maneira nova e melhorada de armazenamento não só irá melhorar o método de armazenamento, mas também ajudar no gerenciamento dos dados OWL de maneira eficiente e melhorando o desempenho das consultas.

2.8 Ambientes Computacionais mais Recentes que Exploram a Abordagem de Competências

Nesta seção falaremos brevemente de alguns ambientes computacionais que utilizam competências.

2.8.1 Uma Extensão de Competências para o Moodle

Em um trabalho realizado por Kalthoum Rezgui, Hédia Mhiri e Khaled Ghédira [6], os autores relatam que vários aspectos relevantes para a gestão de competências estão ausentes no Moodle. Diante disso apresentam o desenvolvimento de uma extensão para o Moodle que fornece aos usuários (criadores, professores, estudantes ou administradores) as capacidades de desenvolvimento e gerenciamento de competências.

A extensão da gestão de competências que propõem é baseada em uma Ontologia de Competências que tem sido desenvolvida em OWL para formalizar os conceitos, evidências e perfil de competência. Essa Ontologia é mostrada na Figura 2.10.

2.8.2 TELOS

O TELOS [29] é basicamente um sistema de coordenação. O TELOS é planejado essencialmente como um conjunto de funcionalidades de coordenação e sincronização para apoiar as interações de pessoas e recursos informatizados que compõem o conhecimento. Ele integra agentes humanos e de computadores usando dois processos básicos: representação semântica de recursos e agregação de recursos. Um objetivo importante é ajudar o sistema a sobreviver à rápida evolução das tecnologias, incorporando modelos independentes de tecnologia de sistema expressos como Ontologias.

O sistema TELOS é capaz de reutilizar Ontologias como “programas conceituais”. Nessa visão, os modelos conceituais não são apenas pré-requisitos para construir o sistema TELOS; eles fazem parte do sistema, como uma das suas camadas mais fundamentais.

2.9 Considerações Finais Sobre o Capítulo

Neste capítulo, foi feito um levantamento da importância dos temas abordados por este referencial teórico. A primeira consideração é relacionada a *Redes Sociais* em ambiente de trabalho. As conferências aqui citadas serviram de base de pesquisa para esse projeto. A medida que foram feitas as pesquisas sobre *Redes Sociais*, foi possível nortear o rumo desse trabalho.

A segunda consideração é sobre o *Friendica*. É de extrema importância citar e explicar as características, de onde surgiu e as pesquisas acadêmicas anteriores a este trabalho. A última consideração está relacionada com Competência: as diferentes modelagens, como é feita a persistências em Base de Dados e os ambientes mais recentes que a explora.

Capítulo 3

Proposta

A proposta deste trabalho é o estudo da viabilidade do desenvolvimento de um Addon, que se acople ao Friendica e contenha funções para a gestão de competências.

Esse capítulo está dividido em duas seções. A primeira apresenta a arquitetura do Addon desenvolvido. A segunda demonstra o protótipo do ponto de vista do usuário, com apresentação das telas do sistema onde os gestores e colaboradores terão acesso.

3.1 Concepção de um Protótipo: Uma Primeira Versão

O Friendica foi desenvolvido utilizando o PHP. Os Addons podem ser desenvolvidos tanto com PHP quanto com Javascript. Para o desenvolvimento do Addon proposto neste trabalho, a linguagem PHP foi escolhida.

Os desenvolvedores da *rede social*, Friendica, disponibilizam uma interface flexível e poderosa para o desenvolvimento dos Addons.

3.1.1 Desenvolvimento

Algumas regras básicas para a escolha do nome do Addon, que será utilizado nos caminhos dos arquivos e nomes de funções, precisam ser seguidas:

1. Não pode conter espaços ou pontuação.
2. Não pode começar com um número.

Neste trabalho, o nome do Addon escolhido será: *competence*. Nas linhas seguintes a variável <addon> terá o seu significado.

Para o funcionamento do Addon, o núcleo do Friendica faz chamadas de eventos, liberando ganchos (hooks), todas as vezes que uma página específica é executada. Os Addons funcionam interceptando estes ganchos, pelo caminho da URL. Com isso, é possível fazer modificações nestas páginas solicitadas.

Os desenvolvedores do Friendica disponibilizam a lista completa de todos os eventos que podem ser consumidos pelo Addon em seu repositório no GitHub [17].

O Addon deve registrar todos os eventos (ganchos) que irão ser utilizados. Para isso, todo Addon deve conter duas funções principais: a de instalação e desinstalação, que irão ser responsáveis por registrar ou anular todos os ganchos utilizados pelo Addon, respectivamente.

Todas as vezes que o Addon for instalado ou reinstalado em um nó do Friendica, os ganchos precisam ser registrados na função com a assinatura: `<addon>_instal()`.

A desinstalação de um complemento anula o registro de todos os gancho registrado com a função: `<addon>_uninstall()`.

Um detalhe importante a cerca destas funções do Addon, é que, sua desinstalação não destrói os dados e a instalação ou reinstalação deve considerar que os dados, utilizados pelo Addon, possam existir.

Elas devem estar presentes em um arquivo `<addon>/<addon>.php`, dentro do diretório `/addons` da instação do nó do Friendica. Ficando as duas funções com a seguinte assinatura:

```
1 function competence_install()
2 function competence_uninstall()
```

Para fazer o registro dos hooks é utilizada a seguinte função, provida pelo núcleo do Friendica:

```
1 Friendica\Core\Hook::register(head, _FILE_, <addon>_head);
```

Onde:

1. A variável 'head' é o nome do gancho que deve ser interceptado (como mencionado anteriormente, cujo a lista completa dos ganchos está disponível para consulta no repositório do Friendica no GitHub).
2. A variável '_FILE_', que deve conter o caminho do arquivo onde se encontra a função de retorno.
3. Por último, a variável '<addon>_head' é nome da função de retorno para todas as vezes que o gancho for executado pelo Friendica.

Prosseguindo com a implementação das funções de install e uninstall, foram criadas o registro para dois hooks: `nav_info` e `profile_tabs`. Sendo que, a primeira é usada para fazer

modificações no menu superior, para prover a pesquisa de Competências e, a segunda, é utilizada para modificar a tela de profile do usuário para adicionar as funcionalidades de cadastro, modificações e remoção de Competências.

Desta forma, o código completo das funções de install e uninstall ficaram da seguinte maneira:

```
1 function <addon>_install() {  
2     register(nav_info , <addon>.php, <addon>_nav_info)  
3     register(profile_tabs , <addon>.php, <addon>_profile_tabs)  
4     log(registered);  
5 }  
6 function <addon>_uninstall() {  
7     unregister(nav_info , <addon>.php, <addon>_nav_info)  
8     unregister(profile_tabs , <addon>.php, <addon>_profile_tabs)  
9     log(unregistered);  
10 }
```

Outras três funções vitais para o funcionamento do Addon, são: init, content e post. A init é chamada todas as vezes que o Addon é inicializado e é responsável por gerir todo o conteúdo que é comum no sistema. O content é responsável por todo o conteúdo que será apresentado em tela para cada iteração do usuário. A post aguarda todas as interações do usuário com o Addon e as intercepta.

Na Figura 3.1, é possível observar todo o fluxo de desenvolvimento do Addon.

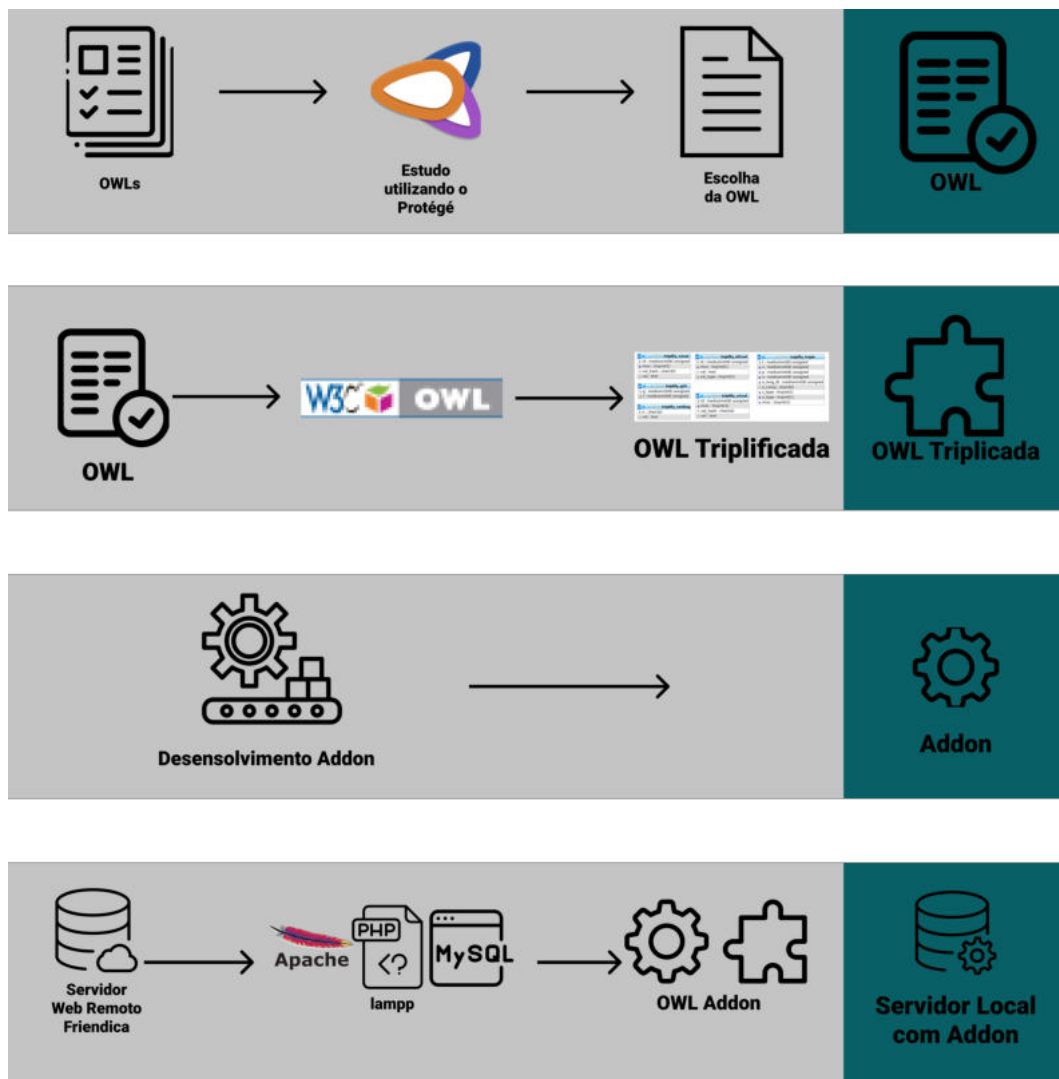


Figura 3.1: Arquitetura aumentada.

3.1.2 Modelagem de Dados

Na Figura 3.2, é possível visualizar o gráfico gerado a partir do protégé com a ontologia utilizada de competências cujo seu código está disponível em anexo.

Para ler o OWL e fazer a persistência dos dados foi utilizado o ARC2, ele triplifica e armazena em tabelas em um banco de dados relacional. Em seguida, foi o utilizado o SPARQL Endpont para manipular as triplas de acordo com as modificações feitas pelos usuários.

3.1.3 Persistência em Base de Dados

O OWL é importado, para o banco de dados chamado triplify criado no phpMyAdmin. Automaticamente, o servidor ARC2, triplifica o OWL e cria as tabelas 3.3 no banco de

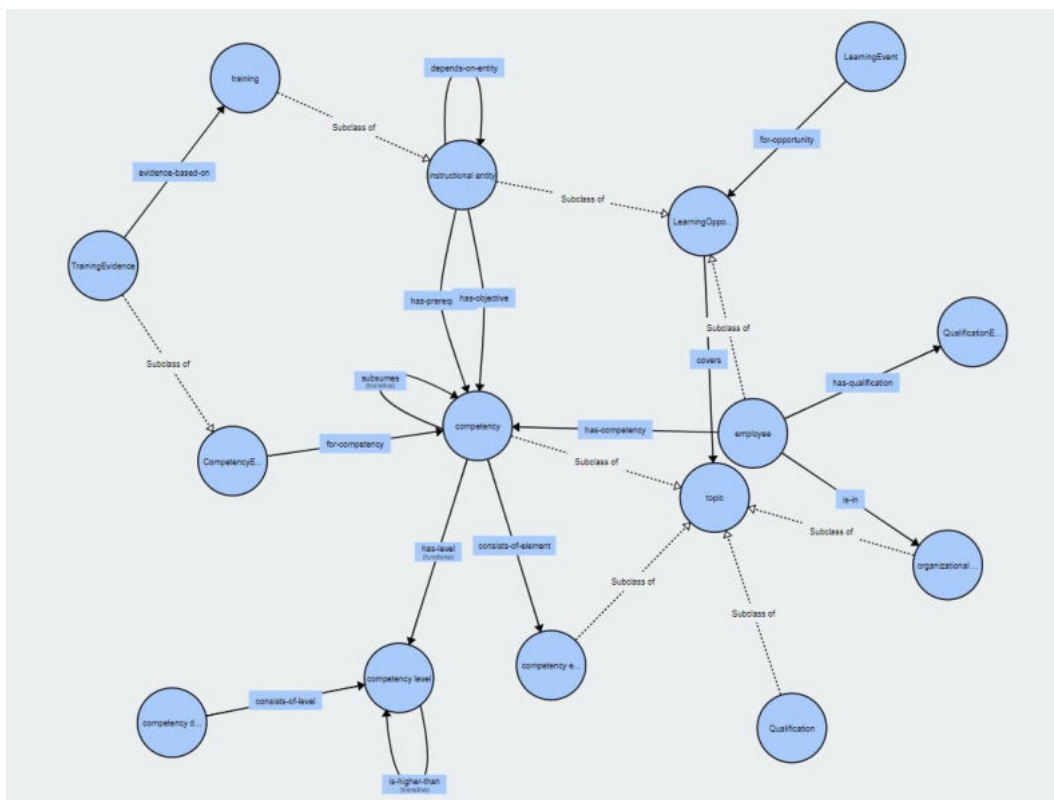


Figura 3.2: Gráfico representativo da Ontologia.

dados e as preenche.

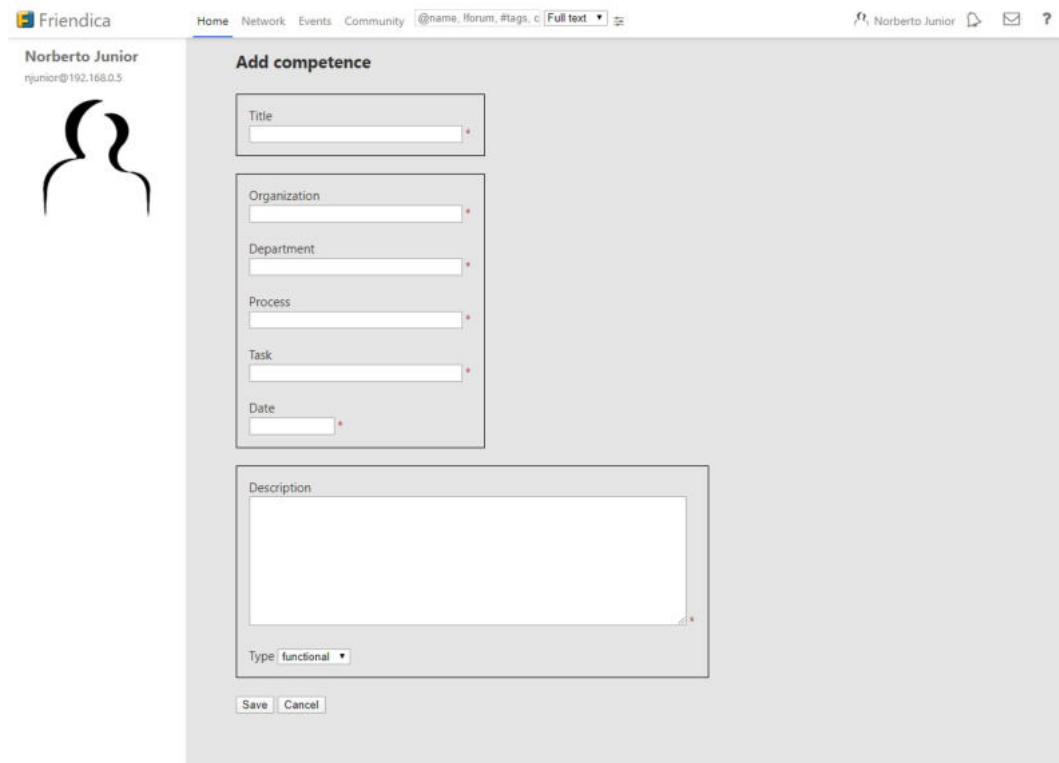
arc2test triplify_s2val id : mediumint(8) unsigned # misc : tinyint(1) val_hash : char(32) val : text	arc2test triplify_id2val id : mediumint(8) unsigned # misc : tinyint(1) val : text val_type : tinyint(1)	arc2test triplify_triple t : mediumint(8) unsigned # s : mediumint(8) unsigned # p : mediumint(8) unsigned # o : mediumint(8) unsigned # o_lang_dt : mediumint(8) unsigned o_comp : char(35) # s_type : tinyint(1) # o_type : tinyint(1) # misc : tinyint(1)
arc2test triplify_g2t g : mediumint(8) unsigned t : mediumint(8) unsigned	arc2test triplify_o2val id : mediumint(8) unsigned # misc : tinyint(1) val_hash : char(32) val : text	
arc2test triplify_setting k : char(32) val : text		

Figura 3.3: Tabelas criadas pelo conversor de Banco de Dados do servidor ARC2.

O repositório triplo é usado para armazenar e consultar os dados RDF. Para acessar e manipular as triplas geradas, foi utilizado o SPARQL EndPoint disponível no servidor ARC2, que implementa o protocolo SPARQL EndPoint. Ele faz a ponte entre o Friendica (Addon) e os dados do OWL.

3.1.4 Inserindo, Alterando, Excluindo (Ator: Usuário Owner)

Na Figura 3.4 é possível visualizar a tela de adição de competências:



The screenshot shows the 'Add competence' form in the Friendica web application. The interface includes a top navigation bar with links for Home, Network, Events, and Community. The user profile 'Norberto Junior' is visible on the left. The form itself is titled 'Add competence' and contains several input fields: Title, Organization, Department, Process, Task, and Date, each with a red asterisk indicating it is required. Below these is a large text area for the Description. At the bottom of the form, there is a 'Type' dropdown menu currently set to 'functional', and 'Save' and 'Cancel' buttons.

Figura 3.4: Adição de competências.

Na Figura 3.5 é possível visualizar a tela de edição de competências:

The screenshot shows the 'Add competence' form in the Friendica web application. The interface includes a top navigation bar with links for Home, Network, Events, and Community, along with a search bar and a user profile icon. On the left, the user's profile is visible, showing the name 'Norberto Junior' and the email 'njunior@192.168.0.5'. The main content area is titled 'Add competence' and contains several input fields: 'Title', 'Organization', 'Department', 'Process', 'Task', and 'Date'. Below these fields is a large text area for 'Description' and a dropdown menu for 'Type' set to 'functional'. At the bottom of the form are 'Save' and 'Cancel' buttons.

Figura 3.5: Edição de competências.

Para deletar, alterar ou adicionar uma competência, os botões ficam na tela de listagem. Na Figura 3.6 é possível visualizar esta tela com 3 competências:

The screenshot shows the 'Competence' list page in the Friendica web application. The interface includes a top navigation bar with links for Home, Network, Events, and Community, along with a search bar and a user profile icon. On the left, the user's profile is visible, showing the name 'Norberto Luz Caires Junior' and the email 'beto_caires@localhost:friendica'. The main content area is titled 'Competence' and contains a list of three competences. Each competence entry includes a 'Título' (Title) and a 'Descrição' (Description). The first competence is titled 'Estrutura de dados' and describes 'Gerenciamento dinâmico da memória. Listas. Pilhas. Filas. Árvores. Ordenação e Pesquisa. Pesquisa em Árvores. Grafos. Análise e aplicação de estruturas avançadas em problemas de programação. Resolução de problemas. Aplicações em problemas ambientais.' The second competence is titled 'Bancos de dados' and describes a detailed list of database topics. The third competence is titled 'CIRCUITOS DIGITAIS' and describes 'Linguagens de descrição de hardware. Álgebra Booleana. Estruturas de Portas Lógicas. Circuitos Combinacionais. Memórias. Circuitos Sequenciais. Conversão A/D e D/A.' At the bottom of the list is a button labeled 'Adicionar competencia'.

Figura 3.6: Listagem de competências.

3.1.5 Consultando (Ator: Gestor)

Na Figura 3.7, é possível visualizar um exemplo de busca de competências. Utilizando o motor de busca já existente do Friendica, foi adicionado a categoria de competências.

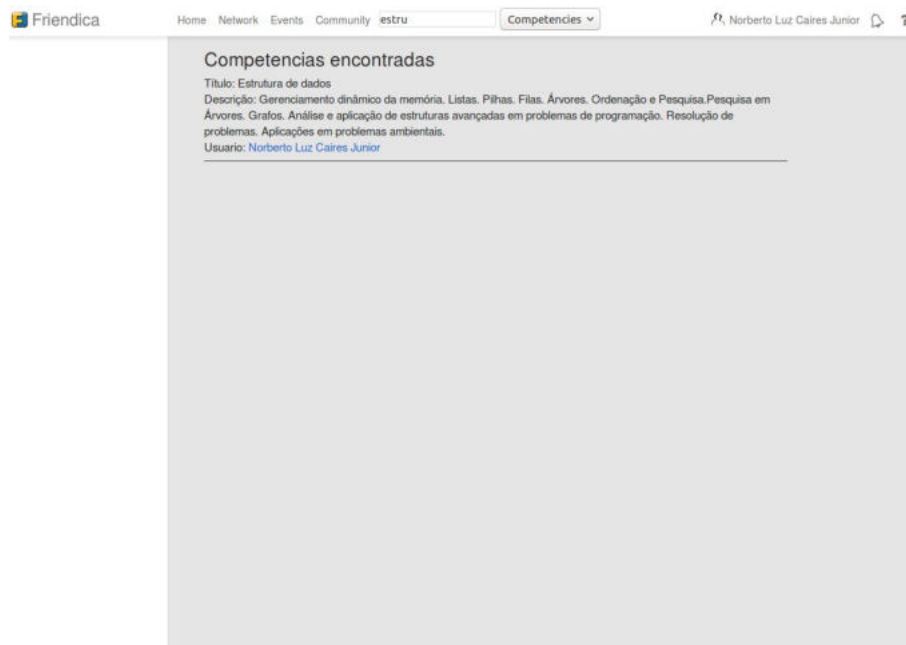


Figura 3.7: Consulta de competências.

É possível buscar e visualizar os dados públicos do perfil de todos os usuários que tenham no nome de suas competências a palavra chave pesquisada. O nome da competência pode ser igual ou contenha no seu subconjunto a palavra chave.

3.2 Análise de Capacidades e Limites

3.2.1 Do Editor

A capacidade do editor é de gerenciar as competências de todos os usuários de um nó do Friendica que contenha a instalação do Addon proposto neste trabalho.

Em seu limite, os dados só estarão disponíveis neste nó. Nenhum outro servidor, que tenha ou não o Addon instalado, será possível visualizar os dados de competências de um outro nó em sua rede.

3.2.2 Do Modelo de Dados / Linguagem de Representação

O modelo de dados que foi proposto pode ser atualizado e melhorado de acordo com as necessidades dos casos de uso. Como o SPARQL EndPoint promete, sua utilização com

grandes quantidades de dados tem um processamento mínimo, pois, faz com que quase todo o processamento fique a cargo do Banco de Dados.

Capítulo 4

Conclusão

4.1 Contribuições

Essa monografia de graduação propôs o desenvolvimento de um protótipo de Addon capaz de fazer a persistência de competências dos usuários na *Rede Social* Friendica e efetuar a busca dos usuários através de palavras chave.

O Friendica e os modelos de competências foram estudados com a finalidade de contribuir para o estudo da integração propostas nos objetivos.

Apesar das dificuldades encontradas, o objetivo de fazer a persistência de competências dos usuário na *Rede Social* foi atingida. O estudo da viabilidade entre *Rede Social* e o Addon foi alcançada, pois o protótipo prover a adição, edição e remoção, assim como a pesquisa de competências aos membros, com a integração do editor de competências ao editor de perfil da Rede.

4.2 Trabalhos Futuros

A seguir são listadas algumas propostas de trabalhos futuros.

1. **Análise da adequação do modelo de competência utilizado para definir a importância das variáveis utilizadas no perfil da Rede.** Os trabalhos apresentados no capítulo 2 dessa monografia apresentam vários campos que compõem competências.
2. **Prover buscas mais sofisticadas, assim como navegação nas competências cadastradas.** Pesquisa não somente por títulos: por comentários; por usuários que apresentem mais de um competência necessária para um projeto, simultaneamente; etc. Uma fácil navegação pelas competências, dando ao gestor uma ampla visualização de todas as competências cadastradas no sistema.

3. **Proposta de agrupamento de usuários por competências, seguindo critérios informado pelo gestor.** Grupos que contenham competências: complementares, iguais, etc. Seguindo regras pré selecionadas pelo gestor.

Referências

- [1] Nóbrega, Germana, Fernando William e Fernanda Lima: *Rumo a uma rede de compromissos em desktop compartilhado: aspectos conceituais e tecnológicos*. páginas 1–4. Simpósio Brasileiro de Sistemas Colaborativos, 2010. ix, 1, 2
- [2] Nóbrega, Germana: *Projeto cloudnet: Apoio à configuração de compromissos profissionais para realização em rede e em nuvem*. Plataforma Carlos Chagas CNPq, outubro 2011. ix, 2, 3
- [3] Kietzmann, Jan, Kristopher Hermkens, Ian McCarthy e Bruno Silvestre: *Social media? get serious! understanding the functional building blocks of social media*. Business Horizons, 54(3):241–251, maio 2011, ISSN 0007-6813. ix, 7, 8, 11
- [4] *Friendica*. <https://github.com/friendica/friendica>, acesso em setembro de 2019. ix, 14, 15, 16
- [5] *professional-learning.eu*. <http://professional-learning.eu/>, acesso em setembro de 2019. ix, 17, 18, 20
- [6] Rezgui, Kalthoum, Hédia Mhiri e Khaled Ghédira: *Extending moodle functionalities with ontology-based competency management*. Procedia Computer Science, 35:570–579, dezembro 2014. ix, 23, 24
- [7] *Sixdegrees*. <https://www.sixdegreesnews.org/what-is-sixdegress>, acesso em setembro de 2019. 1
- [8] *Friendica*. <https://friendi.ca/>, acesso em setembro de 2019. 1, 11
- [9] Boyd, Danah e Nicole Ellison: *Social network sites: Definition, history, and scholarship*. Journal of Computer-Mediated Communication, 13(1):210–230, outubro 2007, ISSN 1083-6101. 6, 7
- [10] Wilson, Bonita: *Social networking gets serious: Social networking moves beyond time-wasting to become a powerful business tool*. Human Resource Management International Digest, 21:6–8, janeiro 2013, ISSN 0967-0734. 9
- [11] DiMicco, Joan, David Millen, Werner Geyer, Casey Dugan, Beth Brownholtz e Michael Muller: *Motivations for social networking at work*. páginas 711–720. Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work, CSCW, janeiro 2008. 10
- [12] Roberts, Bill: *Social media gets strategic*. 59(10):30–34, 2012, ISSN 0967-0734. 10

- [13] Ferris, D: *Special report: social media and hr: from neophytes to socialites*. 91(6):32–4, 2012. 10
- [14] *Statement*. https://youtu.be/T_X-Qoh_mw, acesso em setembro de 2019. 11
- [15] *United breaks guitars*. <https://youtu.be/5YGc4z0qozo>, acesso em setembro de 2019. 11
- [16] *Xampp*. <https://www.apachefriends.org>, acesso em setembro de 2019. 11
- [17] *Friendica addons*. <https://github.com/friendica/friendica-addons>, acesso em setembro de 2019. 12, 27
- [18] Nagulendra, Sayooran e Julita Vassileva: *Providing awareness, explanation and control of personalized filtering in a social networking site*. Information Systems Frontiers, 18(1):145–158, junho 2015. 12
- [19] Onrec: *Competency profiling fits the bill*. 2007. <http://www.onrec.com/news/news-archive/competency-profiling-fits-the-bill>, acesso em setembro de 2019. 16
- [20] Cheetham, Graham e Geoff Chivers: *Professions, competence and informal learning*. Journal of Interprofessional Care, 20, janeiro 2005, ISSN 1356-1820. 16
- [21] Cheetham, Graham e Geoffrey Chivers: *Professions, competence and informal learning*. Journal of Interprofessional Care, 20, janeiro 2005. 17
- [22] Tobias, Ley: *Organizational competence management - a competence performance approach*. In: 6th International Conference on Knowledge Magement (I-KNOW 06), Graz, Austria, 01. 17
- [23] Hamel, Gary e Coimbatore Prahalad: *Competing for the future*. Harvard business review, 72(4):122–128, 1994. 17
- [24] Sampson, Demetrios e Demetrios Fytros: *Competence models in technology-enhanced competence-based learning*. Em *Handbook on information technologies for education and training*, volume 59, páginas 155–177. Heidelberg: Springer, janeiro 2008. 19
- [25] Smith, Michael, Chris Welty e Deborah Mcguinness: *Owl web ontology language guide*. <http://www.w3.org/TR/2004/REC-owl-guide-20040210/>, acesso em setembro de 2019. 19
- [26] Alamri, Abdullah: *The relational database layout to store ontology knowledge base*. Em *2012 International Conference on Information Retrieval & Knowledge Management*, páginas 74–81. março 2012. 21, 22
- [27] Breitman, Karin, Marco Casanova e Walt Truszkowski: *Semantic web: concepts, technologies and applications*. NASA Monographs in Systems and Software Engineering, 1, janeiro 2007. 21

- [28] *Arc - rdf and sparql for php developers*. <http://www.semanticfocus.com/blog/entry/title/arc-rdf-and-sparql-for-php-developers/>, acesso em setembro de 2019. 22
- [29] Paquette, Gilbert: *An ontology and a software framework for competency modeling and management*. Journal of Educational Technology & Society, 10(3):1–21, julho 2007. 25

Anexo I

Instalação do Friendica

I.1 Instalação Friendica

Em uma máquina com Ubuntu 16.04 instalado e sem a instalação do PHP, Apache ou MySQL. Iremos instalar o Xampp, pois ele já provê: banco de dados MySQL, Apache e interpretador PHP. Logo em seguida mostramos a instalação do Friendica e do Addon.

I.1.1 Instalação do Xampp

Faça o download da versão mais recente do Xamp, quando este tutorial foi feito estava na versão 7.2.6, para Linux.

```
1 https://www.apachefriends.org/download.html
```

Abra o terminal e vá para pasta que o arquivo foi salvo:

```
1 cd [pasta de download]
```

Dê permissão de leitura e escrita:

```
1 sudo chmod 777 [nome do arquivo]
```

Execute o arquivo com:

```
1 sudo ./[nome do arquivo]
```

Siga os passos para instalação na nova janela de diálogos que foi aberta.

Aqui, assumiremos que a instalação do Xampp foi feita na pasta: /opt/lampp, como mostrado na Figura I.1.

Após a instalação, certifique-se que o MySQL, Apache estão rodando e digite o seguinte comando para que o PHP do Xampp seja o principal:

```
1 sudo ln -s /opt/lampp/bin/php /usr/bin/php
```

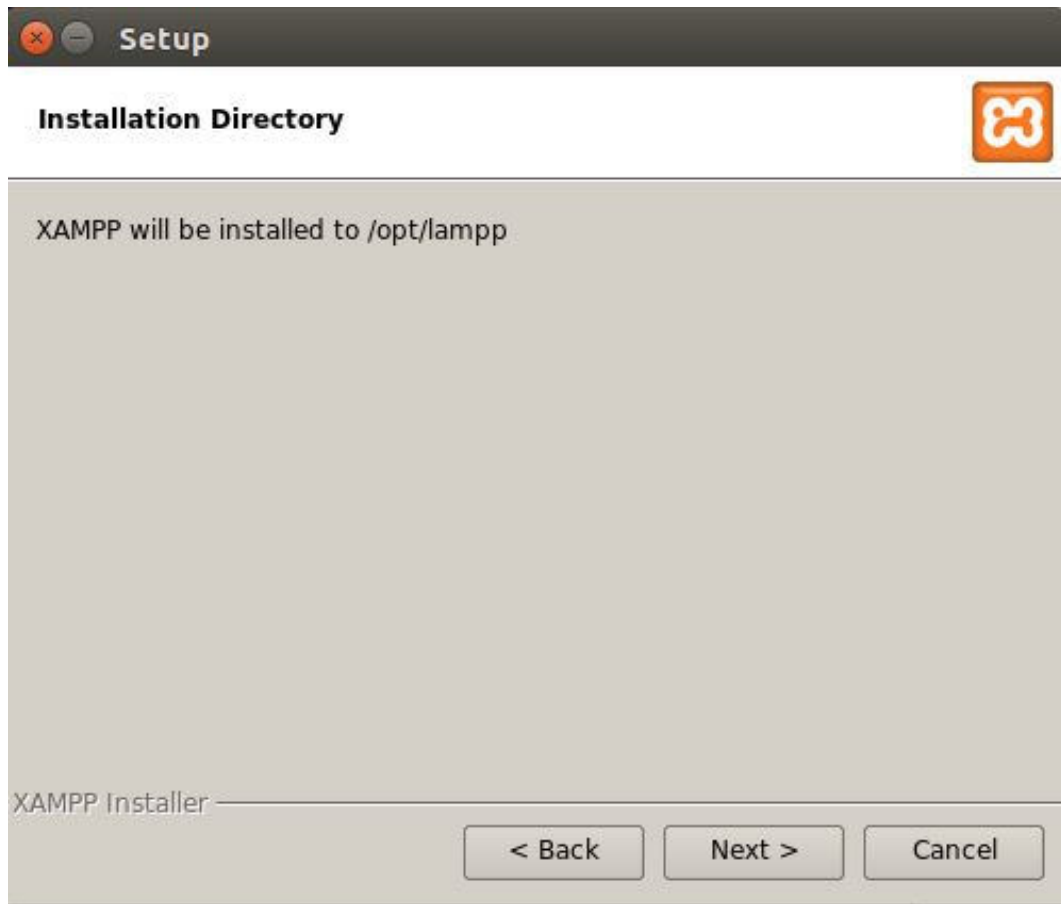


Figura I.1: Instalando Xampp.

A Figura I.2 mostra que o Xampp foi instalado e que o MySQL, ProFTPD e Apache estão rodando corretamente.

I.1.2 Instalação do Friendica

Para começar, iremos fazer o download do Friendica. Iremos clonar o projeto por linha de comando na pasta htdocs do xampp:

```
1 sudo git clone https://github.com/friendica/friendica.git /opt/  
    lampp/htdocs/friendica
```

Daremos permissão de leitura e escrita na pasta, utilize o -R para ser recursivo, assim todas as subpastas terão a mesma permissão:

```
1 sudo chmod -R 777 /opt/lampp/htdocs/friendica/
```

Entre na pasta que acabamos de criar clonando o Friendica com o seguinte comando:

```
1 cd /opt/lampp/htdocs/friendica
```

Rode a linha de comando a seguir para iniciar a instalação:

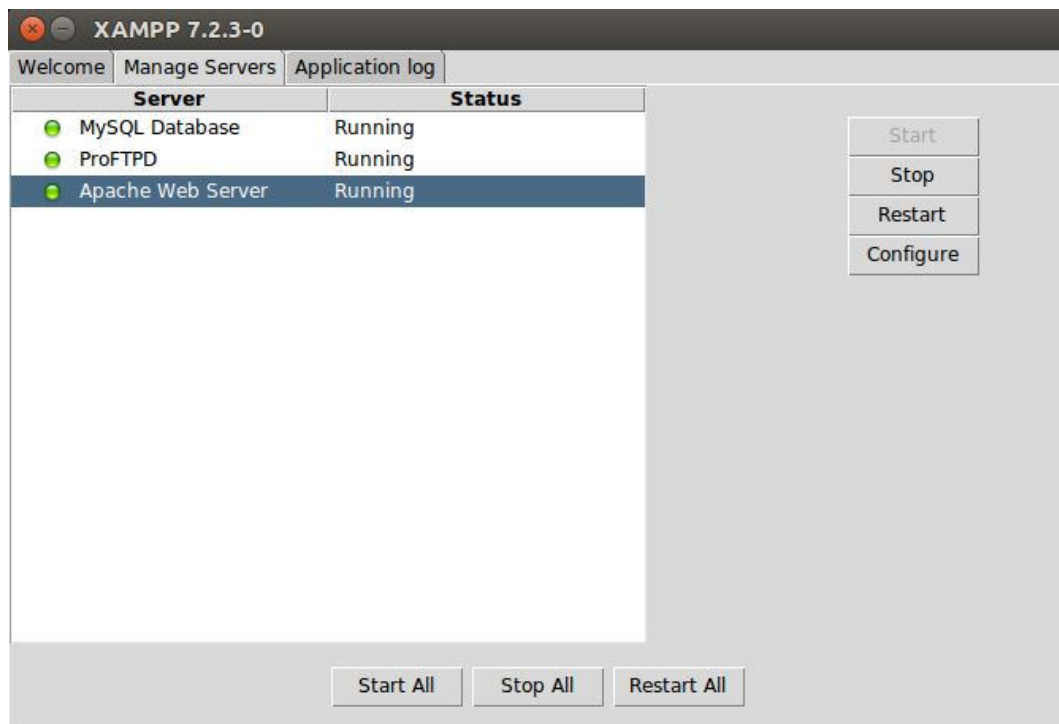


Figura I.2: Xampp Instalado.

```
1 php bin/composer.phar install
```

Importante: não execute este comando como root/super usuário.

Abra um navegador e siga os passos da instalação em:

```
1 http://localhost/friendica
```

```
teste@teste-VirtualBox:/opt/lampp/htdocs/friendica$ sudo chmod -R 777 /opt/lampp/htdocs/friendica/
[sudo] password for teste:
teste@teste-VirtualBox:/opt/lampp/htdocs/friendica$ php bin/composer.phar install
Loading composer repositories with package information
Installing dependencies (including require-dev) from lock file
Package operations: 37 installs, 0 updates, 0 removals
- Installing fxp/composer-asset-plugin (v1.4.2): Downloading (100%)
- Installing asika/simple-console (1.0.3): Downloading (100%)
- Installing bower-asset/chart-js (v2.7.1): Downloading (100%)
- Installing bower-asset/base64 (1.0.1): Downloading (100%)
- Installing bower-asset/perfect-scrollbar (0.6.16): Downloading (100%)
- Installing rapidwebitd/rw-file-cache (v1.2.5): Downloading (100%)
- Installing psr/cache (1.0.1): Downloading (100%)
- Installing rapidwebitd/rw-file-cache-psr-6 (v1.0.0): Downloading (100%)
- Installing paragonie/random_compat (v2.0.11): Downloading (100%)
- Installing paragonie/sodium_compat (v1.6.0): Downloading (100%)
- Installing paragonie/constant-time-encoding (v1.0.2): Downloading (100%)
- Installing psr/http-message (1.0.1): Downloading (100%)
- Installing guzzlehttp/psr7 (1.4.2): Downloading (100%)
- Installing guzzlehttp/promises (v1.3.1): Downloading (100%)
- Installing guzzlehttp/guzzle (6.3.0): Downloading (100%)
- Installing paragonie/certainty (v1.0.2): Downloading (100%)
- Installing divineomega/password-exposed (v2.4.0): Downloading (100%)
- Installing ezyang/htmlpurifier (v4.7.0): Downloading (100%)
- Installing league/html-to-markdown (4.4.1): Downloading (100%)
- Installing lightopenid/lightopenid (dev-master 9d42dc2): Downloading (100%)
- Installing michelf/php-markdown (1.8.0): Downloading (100%)
- Installing mobiledetect/mobiledetectlib (2.8.30): Downloading (100%)
- Installing npm-asset/cropperjs (1.2.2): Downloading (100%)
- Installing npm-asset/moment (2.20.1): Downloading (100%)
- Installing npm-asset/jquery (2.2.4): Downloading (100%)
- Installing npm-asset/fullcalendar (3.8.2): Downloading (100%)
- Installing npm-asset/jgrowl (1.4.6): Downloading (100%)
- Installing npm-asset/jquery-colorbox (1.6.4): Downloading (100%)
- Installing npm-asset/php-date-formatter (1.3.4): Downloading (100%)
- Installing npm-asset/jquery-mousewheel (3.1.13): Downloading (100%)
- Installing npm-asset/jquery-datetimepicker (2.5.17): Downloading (100%)
- Installing pear/console_getopt (v1.4.1): Downloading (100%)
- Installing pear/pear_exception (v1.0.0): Downloading (100%)
- Installing pear/pear-core-minimal (v1.10.3): Downloading (100%)
- Installing pear/text-highlighter (dev-master 2ccac2d): Downloading (100%)
- Installing pear/text-language-detect (v1.0.0): Downloading (100%)
- Installing smarty/smarty (v3.1.31): Downloading (100%)
paragonie/random_compat suggests installing ext-libsodium (Provides a modern crypto API that can be used to generate random bytes.)
paragonie/sodium_compat suggests installing ext-libsodium (PHP < 7.0: Better performance, password hashing (Argon2i), secure memory management (memzero), and better security.)
paragonie/sodium_compat suggests installing ext-sodium (PHP >= 7.0: Better performance, password hashing (Argon2i), secure memory management (memzero), and better security.)
guzzlehttp/guzzle suggests installing psr/log (Required for using the Log middleware)
Generating optimized autoload files
teste@teste-VirtualBox:/opt/lampp/htdocs/friendica$
```

Figura I.3: Instalado Friendica terminal.

Neste ponto, a página deve mostrar uma série de requisitos e se os mesmo estão sendo suportados.

Certifique-se de que esteja atendendo todos os requisitos, se sim, no final da página teremos um botão de Next, como no exemplo da Figura I.4.

Na próxima página o Friendica pede acesso ao banco de dados.

Acesse:

```
1 https://localhost/phpmyadmin
```

Crie um banco com o nome friendica (ou outro nome que desejar) e a codificação utf8mb4 e rode o sql do arquivo:

```
1 /opt/lampp/htdocs/friendica/database.sql
```

Assim, teremos um banco com todas as tabelas que o Friendica necessita.

Volte em:

```
1 http://localhost/friendica
```

e preencha como é mostrado na Figura I.5.

Database login: name e password pode ser um usuário do banco que desejar. Clique em Submit.

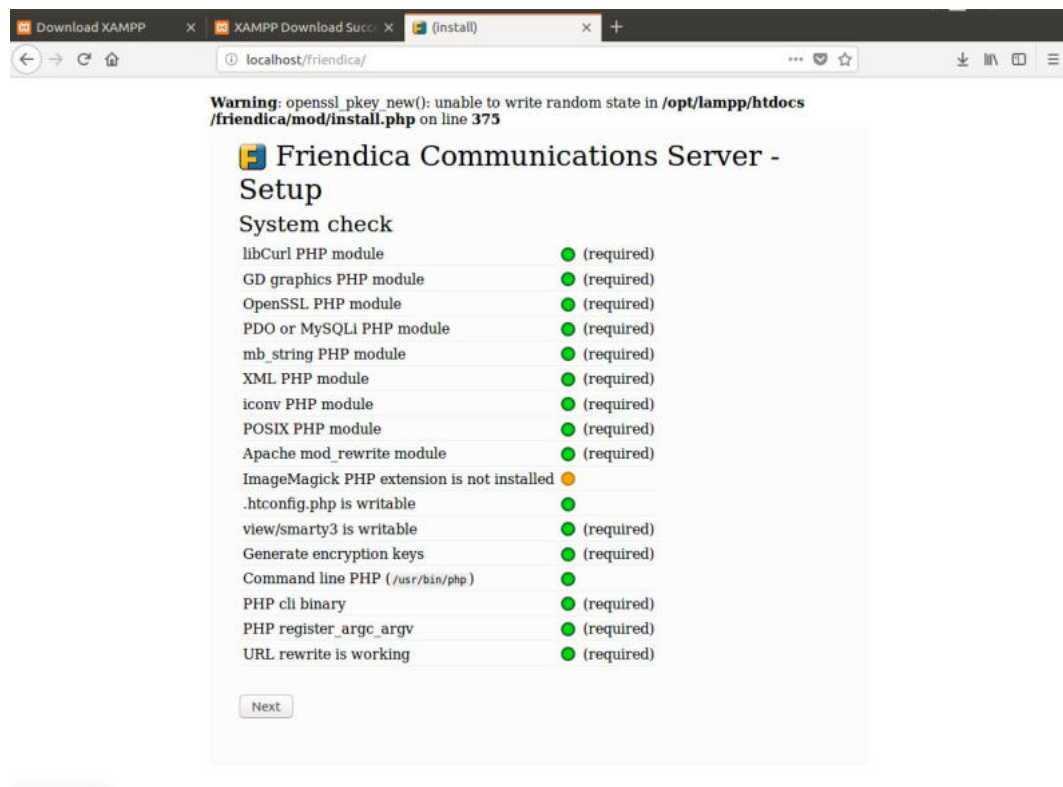


Figura I.4: Instalado Friendica - Pré-requisitos.

Escolha as opções de língua, time zone e o e-mail do administrador. Um exemplo é mostrado na Figura I.6.

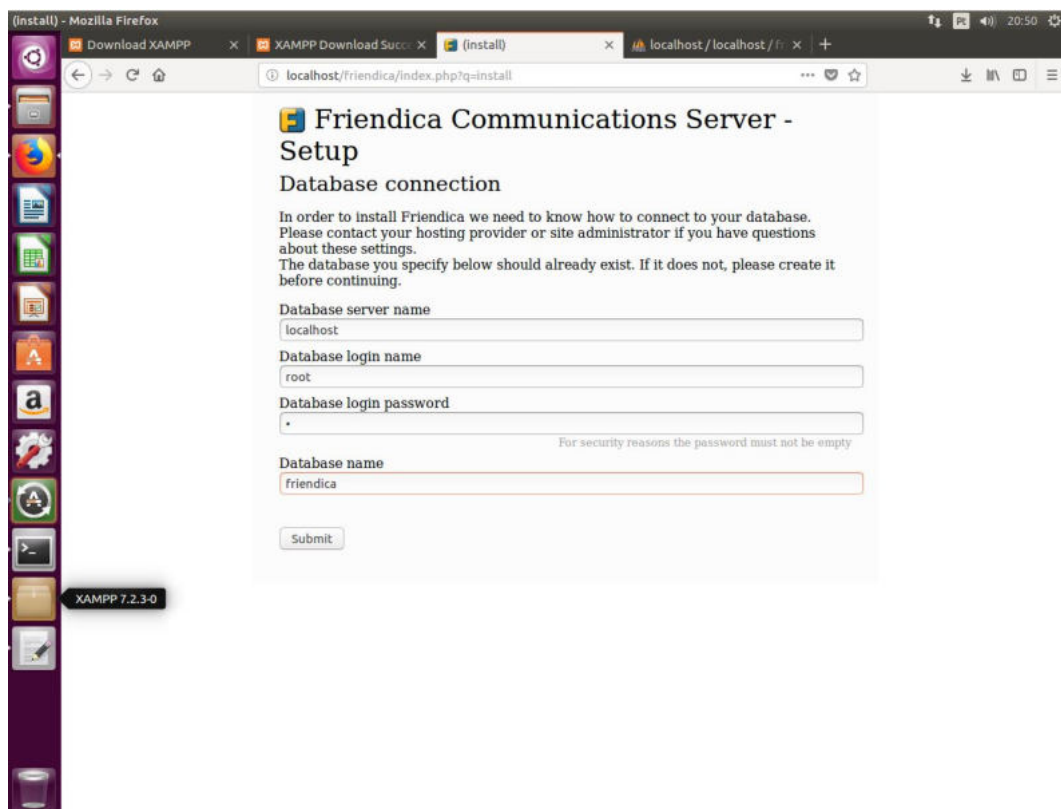


Figura I.5: Instalado Friendica - Banco de dados.

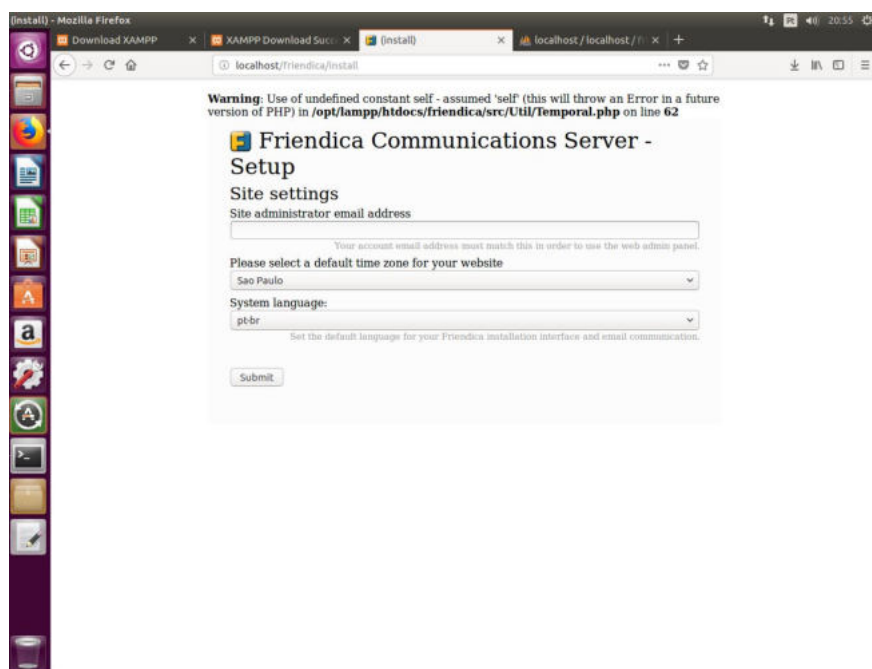


Figura I.6: Instalado Friendica - Configuração.

Clique em Submit e deverá aparecer a seguinte tela I.7.

Comece a utilizar o Friendica localmente em: <http://localhost/friendica>.

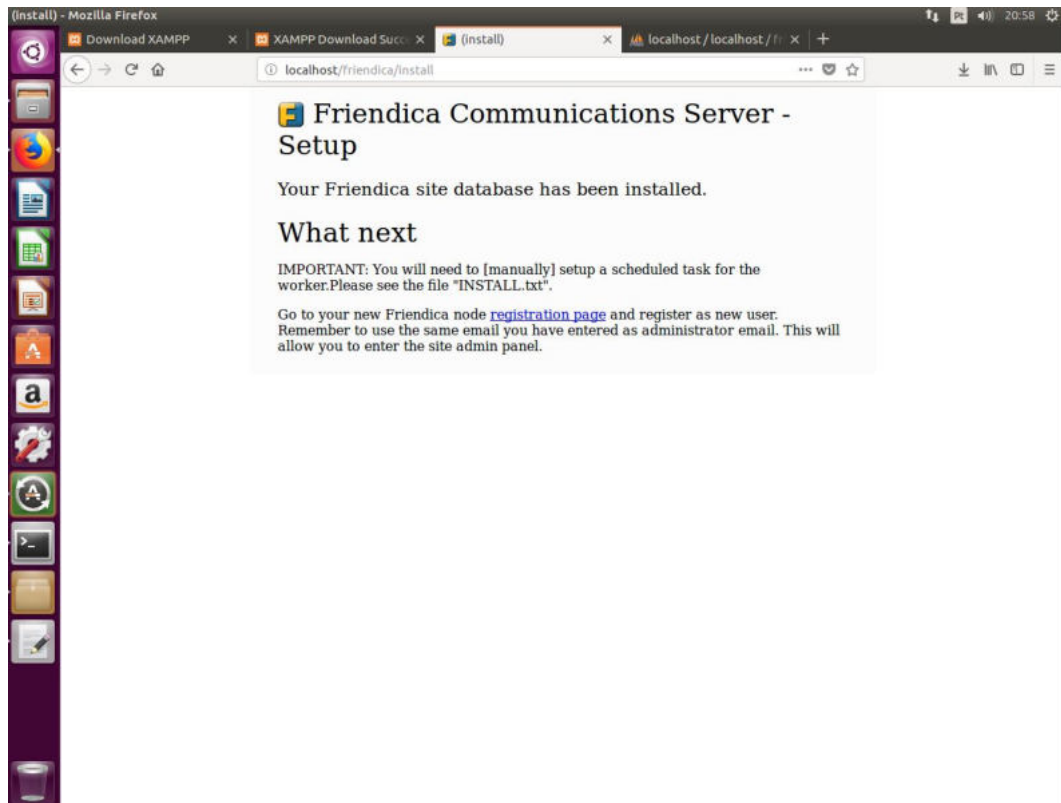


Figura I.7: Instalação concluída.

I.2 Instalação Addon

Clone o projeto por linha de comando na pasta addons do Friendica:

```
1 sudo git clone https://github.com/norbertocaires/competence.git  
/opt/lampp/htdocs/friendica/addon/competence
```

Crie um banco com o arc2 e importe a OWL com o seguinte comando:

```
1 ./cli.php "LOAD <file:///opt/lampp/htdocs/friendica/addon/  
competence/arc2-starter-pack/owl.owl>"
```

Para finalizar, vá em configuração com o perfil de Admin do Friendica, clique em Addons e marque o Addon Competencie para utilização.

Anexo II

OWL

```
1 <?xml version="1.0"?>
2 <rdf:RDF
3   xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"
4   xmlns:swrl="http://www.w3.org/2003/11/swrl#"
5   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
6   xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
7   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
8   xmlns:owl="http://www.w3.org/2002/07/owl#"
9   xmlns:daml="http://www.daml.org/2001/03/daml+oil#"
10  xmlns:dc="http://purl.org/dc/elements/1.1/"
11  xmlns="http://www.professional-learning.eu/ontologies/
    competence.owl#"
12  xml:base="http://www.professional-learning.eu/ontologies/
    competence.owl">
13  <owl:Ontology rdf:about="">
14    <dc:creator rdf:datatype="http://www.w3.org/2001/XMLSchema#
      string"
15    >Andreas Schmidt (Andreas.Schmidt@fzi.de)</dc:creator>
16    <owl:versionInfo rdf:datatype="http://www.w3.org/2001/
      XMLSchema#string"
17    >Version 1.0, 25.6.2006</owl:versionInfo>
18    <owl:imports rdf:resource="http://www.w3.org/2003/11/swrl"/>
19    <dc:contributor rdf:datatype="http://www.w3.org/2001/
      XMLSchema#string"
20    >Christine Kunzmann</dc:contributor>
21    <owl:imports rdf:resource="http://protege.stanford.edu/
      plugins/owl/dc/protege-dc.owl"/>
```

```

22 <dc:publisher xml:lang="en">FZI Research Center for
    Information Technologies, Information Process
23 Engineering, Karlsruhe, Germany - http://www.fzi.de</dc:
    publisher>
24 <dc:rights rdf:datatype="http://www.w3.org/2001/XMLSchema#
    string"
25 >(c) Andreas Schmidt 2006</dc:rights>
26 <dc:title xml:lang="en">Reference Ontology for Competency
    Management and Competency-Oriented
27 Learning Support</dc:title>
28 <owl:imports rdf:resource="http://www.w3.org/2003/11/swrlb"/>
29 <dc:description xml:lang="en">This ontology represents a top-
    level ontology for competence management
30 application and competency-oriented learning support systems.
    It has been
31 developed in the Information Process Engineering group at FZI
    Research
32 Center for Information Technologies, Karlsruhe, Germany</dc:
    description>
33 </owl:Ontology>
34 <owl:Class rdf:ID="Task">
35   <rdfs:subClassOf>
36     <owl:Class rdf:ID="OrganizationalEntity"/>
37   </rdfs:subClassOf>
38 </owl:Class>
39 <owl:Class rdf:ID="LanguageProficiencyScale">
40   <rdfs:label xml:lang="de">Sprachleistungsskala</rdfs:label>
41   <rdfs:label xml:lang="en">language proficiency scale</rdfs:
    label>
42   <rdfs:subClassOf>
43     <owl:Class rdf:ID="CompetencyScale"/>
44   </rdfs:subClassOf>
45 </owl:Class>
46 <owl:Class rdf:about="#OrganizationalEntity">
47   <rdfs:label xml:lang="en">organizational entity</rdfs:label>
48   <rdfs:subClassOf>
49     <owl:Class rdf:ID="Topic"/>
50   </rdfs:subClassOf>
51   <rdfs:label xml:lang="de">Organisationseinheit</rdfs:label>
52 </owl:Class>

```

```

53 <owl:Class rdf:ID="SelfAssessment">
54   <rdfs:subClassOf>
55     <owl:Class rdf:ID="CompetencyEvidence"/>
56   </rdfs:subClassOf>
57   <rdfs:label xml:lang="en">self assessment</rdfs:label>
58   <rdfs:label xml:lang="de">Selbstbewertung</rdfs:label>
59 </owl:Class>
60 <owl:Class rdf:ID="ProcessActivity">
61   <rdfs:subClassOf rdf:resource="#OrganizationalEntity"/>
62 </owl:Class>
63 <owl:Class rdf:ID="Qualification">
64   <rdfs:label xml:lang="en">Qualification</rdfs:label>
65   <rdfs:subClassOf>
66     <owl:Class rdf:about="#Topic"/>
67   </rdfs:subClassOf>
68   <rdfs:label xml:lang="de">Qualifikation</rdfs:label>
69 </owl:Class>
70 <owl:Class rdf:ID="QualificationEvidence"/>
71 <owl:Class rdf:ID="CompetencyLevel">
72   <rdfs:label xml:lang="de">Kompetenzniveau</rdfs:label>
73   <rdfs:label xml:lang="en">competency level</rdfs:label>
74 </owl:Class>
75 <owl:Class rdf:about="#CompetencyScale">
76   <rdfs:label xml:lang="en">competency dimension</rdfs:label>
77 </owl:Class>
78 <owl:Class rdf:ID="TrainingEvidence">
79   <rdfs:subClassOf rdf:resource="#CompetencyEvidence"/>
80 </owl:Class>
81 <owl:Class rdf:about="#Topic">
82   <synonym xml:lang="de">subject</synonym>
83   <rdfs:label xml:lang="en">topic</rdfs:label>
84   <rdfs:label xml:lang="de">Thema</rdfs:label>
85 </owl:Class>
86 <owl:Class rdf:ID="Training">
87   <rdfs:label xml:lang="en">training</rdfs:label>
88   <rdfs:subClassOf>
89     <owl:Class rdf:ID="InstructionalEntity"/>
90   </rdfs:subClassOf>
91   <rdfs:label xml:lang="de">Schulung</rdfs:label>
92 </owl:Class>

```

```

103 <owl:Class rdf:about="#InstructionalEntity">
104   <rdfs:label xml:lang="en">instructional entity</rdfs:label>
105   <rdfs:label xml:lang="de">Lehreinheit</rdfs:label>
106   <rdfs:subClassOf>
107     <owl:Class rdf:ID="LearningOpportunity"/>
108   </rdfs:subClassOf>
109 </owl:Class>
110 <owl:Class rdf:ID="Ability">
111   <rdfs:subClassOf>
112     <owl:Class rdf:ID="CompetencyElement"/>
113   </rdfs:subClassOf>
114   <rdfs:label xml:lang="de">Fähigkeit</rdfs:label>
115   <rdfs:label xml:lang="en">ability</rdfs:label>
116 </owl:Class>
117 <owl:Class rdf:ID="Observation">
118   <rdfs:subClassOf rdf:resource="#CompetencyEvidence"/>
119   <rdfs:label xml:lang="en">observation</rdfs:label>
120   <rdfs:label xml:lang="de">Beobachtung</rdfs:label>
121 </owl:Class>
122 <owl:Class rdf:ID="Skill">
123   <rdfs:label xml:lang="en">skill</rdfs:label>
124   <rdfs:label xml:lang="de">Fertigkeit</rdfs:label>
125   <rdfs:subClassOf>
126     <owl:Class rdf:about="#CompetencyElement"/>
127   </rdfs:subClassOf>
128 </owl:Class>
129 <owl:Class rdf:ID="LearningEvent"/>
130 <owl:Class rdf:about="#CompetencyElement">
131   <rdfs:label xml:lang="de">Kompetenzbaustein</rdfs:label>
132   <rdfs:label xml:lang="en">competency element</rdfs:label>
133   <rdfs:subClassOf rdf:resource="#Topic"/>
134 </owl:Class>
135 <owl:Class rdf:ID="Employee">
136   <synonym xml:lang="en">learner</synonym>
137   <rdfs:subClassOf>
138     <owl:Class rdf:about="#LearningOpportunity"/>
139   </rdfs:subClassOf>
140   <rdfs:label xml:lang="en">employee</rdfs:label>
141   <rdfs:label xml:lang="de">Mitarbeiter</rdfs:label>
142 </owl:Class>

```

```

133 <owl:Class>
134   <owl:unionOf rdf:parseType="Collection">
135     <owl:Class rdf:about="#CompetencyEvidence"/>
136     <owl:Class rdf:about="#LearningEvent"/>
137   </owl:unionOf>
138 </owl:Class>
139 <owl:Class rdf:ID="Competency">
140   <rdfs:label xml:lang="en">competency</rdfs:label>
141   <rdfs:subClassOf rdf:resource="#Topic"/>
142   <synonym xml:lang="en">competence</synonym>
143   <rdfs:label xml:lang="de">Kompetenz</rdfs:label>
144 </owl:Class>
145 <owl:Class rdf:ID="Process">
146   <rdfs:subClassOf rdf:resource="#OrganizationalEntity"/>
147 </owl:Class>
148 <owl:Class rdf:about="#LearningOpportunity">
149   <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#
150     string"
151   >&lt;p style="margin-top: 0">
152     &lt;/p></rdfs:comment>
153   <rdfs:label xml:lang="de">Lerngelegenheit</rdfs:label>
154 </owl:Class>
155 <owl:Class rdf:ID="OrganizationalUnit">
156   <rdfs:subClassOf rdf:resource="#OrganizationalEntity"/>
157 </owl:Class>
158 <owl:Class rdf:ID="Knowledge">
159   <rdfs:label xml:lang="en">knowledge</rdfs:label>
160   <rdfs:label xml:lang="de">Kenntnis</rdfs:label>
161   <rdfs:subClassOf rdf:resource="#CompetencyElement"/>
162 </owl:Class>
163 <owl:Class rdf:ID="LearningObject">
164   <rdfs:label xml:lang="en">learning object</rdfs:label>
165   <rdfs:label xml:lang="de">Lernobjekt</rdfs:label>
166   <rdfs:subClassOf rdf:resource="#InstructionalEntity"/>
167 </owl:Class>
168 <owl:ObjectProperty rdf:ID="has-competency">
169   <rdfs:range rdf:resource="#Competency"/>
170   <rdfs:subPropertyOf>
171     <owl:ObjectProperty rdf:ID="covers"/>

```

```

172     </rdfs:subPropertyOf>
173     <rdfs:domain rdf:resource="#Employee"/>
174 </owl:ObjectProperty>
175 <owl:ObjectProperty rdf:ID="for-opportunity">
176     <rdfs:range rdf:resource="#LearningOpportunity"/>
177     <rdfs:domain rdf:resource="#LearningEvent"/>
178 </owl:ObjectProperty>
179 <owl:ObjectProperty rdf:about="#covers">
180     <rdfs:domain rdf:resource="#LearningOpportunity"/>
181     <rdfs:range rdf:resource="#Topic"/>
182 </owl:ObjectProperty>
183 <owl:ObjectProperty rdf:ID="consists-of-element">
184     <rdfs:range rdf:resource="#CompetencyElement"/>
185     <rdfs:domain rdf:resource="#Competency"/>
186 </owl:ObjectProperty>
187 <owl:ObjectProperty rdf:ID="consists-of-level">
188     <rdfs:range rdf:resource="#CompetencyLevel"/>
189     <rdfs:domain rdf:resource="#CompetencyScale"/>
190 </owl:ObjectProperty>
191 <owl:ObjectProperty rdf:ID="depends-on-entity">
192     <rdfs:range rdf:resource="#InstructionalEntity"/>
193     <rdfs:domain rdf:resource="#InstructionalEntity"/>
194 </owl:ObjectProperty>
195 <owl:ObjectProperty rdf:ID="for-competency">
196     <rdfs:range rdf:resource="#Competency"/>
197     <rdfs:domain rdf:resource="#CompetencyEvidence"/>
198 </owl:ObjectProperty>
199 <owl:ObjectProperty rdf:ID="has-objective">
200     <rdfs:domain rdf:resource="#InstructionalEntity"/>
201     <rdfs:subPropertyOf rdf:resource="#covers"/>
202     <rdfs:range rdf:resource="#Competency"/>
203 </owl:ObjectProperty>
204 <owl:ObjectProperty rdf:ID="is-in">
205     <rdfs:range rdf:resource="#OrganizationalEntity"/>
206     <rdfs:domain rdf:resource="#Employee"/>
207 </owl:ObjectProperty>
208 <owl:ObjectProperty rdf:ID="is-composed-of">
209     <rdfs:range rdf:resource="#Competency"/>
210     <rdfs:domain>
211         <owl:Class>

```

```

212         <owl:unionOf rdf:parseType="Collection">
213             <owl:Class rdf:about="#Qualification"/>
214             <owl:Class rdf:about="#Competency"/>
215         </owl:unionOf>
216     </owl:Class>
217 </rdfs:domain>
218 </owl:ObjectProperty>
219 <owl:ObjectProperty rdf:ID="subsumes">
220     <rdfs:range rdf:resource="#Competency"/>
221     <rdfs:domain rdf:resource="#Competency"/>
222     <rdfs:type rdf:resource="http://www.w3.org/2002/07/owl#
        TransitiveProperty"/>
223 </owl:ObjectProperty>
224 <owl:ObjectProperty rdf:ID="evidence-based-on">
225     <rdfs:range rdf:resource="#Training"/>
226     <rdfs:domain rdf:resource="#TrainingEvidence"/>
227 </owl:ObjectProperty>
228 <owl:ObjectProperty rdf:ID="has-prerequisite">
229     <rdfs:domain rdf:resource="#InstructionalEntity"/>
230     <rdfs:range rdf:resource="#Competency"/>
231 </owl:ObjectProperty>
232 <owl:ObjectProperty rdf:ID="has-level">
233     <rdfs:range rdf:resource="#CompetencyLevel"/>
234     <rdfs:type rdf:resource="http://www.w3.org/2002/07/owl#
        FunctionalProperty"/>
235     <rdfs:domain rdf:resource="#Competency"/>
236 </owl:ObjectProperty>
237 <owl:ObjectProperty rdf:ID="has-qualification">
238     <rdfs:range rdf:resource="#QualificationEvidence"/>
239     <rdfs:domain rdf:resource="#Employee"/>
240 </owl:ObjectProperty>
241 <owl:TransitiveProperty rdf:ID="is-higher-than">
242     <rdfs:domain rdf:resource="#CompetencyLevel"/>
243     <rdfs:range rdf:resource="#CompetencyLevel"/>
244     <rdfs:type rdf:resource="http://www.w3.org/2002/07/owl#
        ObjectProperty"/>
245 </owl:TransitiveProperty>
246 <owl:FunctionalProperty rdf:ID="confidence">
247     <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#
        float"/>

```

```

248     <rdfs:domain rdf:resource="#Observation"/>
249     <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#
        DatatypeProperty"/>
250 </owl:FunctionalProperty>
251 <owl:InverseFunctionalProperty rdf:ID="has-evidence">
252     <rdfs:range>
253         <owl:Class>
254             <owl:unionOf rdf:parseType="Collection">
255                 <owl:Class rdf:about="#CompetencyEvidence"/>
256                 <owl:Class rdf:about="#QualificationEvidence"/>
257             </owl:unionOf>
258         </owl:Class>
259     </rdfs:range>
260     <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#
        ObjectProperty"/>
261     <rdfs:domain rdf:resource="#Employee"/>
262 </owl:InverseFunctionalProperty>
263 <owl:AnnotationProperty rdf:ID="has-scale">
264     <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#
        ObjectProperty"/>
265     <rdfs:range rdf:resource="#CompetencyScale"/>
266 </owl:AnnotationProperty>
267 <owl:AnnotationProperty rdf:ID="synonym">
268     <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#
        string"/>
269     <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#
        DatatypeProperty"/>
270 </owl:AnnotationProperty>
271 </rdf:RDF>
272
273 <!-- Created with Protege (with OWL Plugin 2.2, Build 322) http:
    //protege.stanford.edu -->

```